

GFK-0467L-P

[Buy GE Fanuc Series 90-30 NOW!](#)

GE Fanuc Manual Series 90-30

Zestaw instrukcji dla jednostek centralnych
sterownik w serii 90 -30/20/Micro

1-800-360-6802
sales@pdfsupply.com



GE Fanuc Automation

Sterowniki programowalne

***Zestaw instrukcji dla jednostek centralnych
sterowników serii 90™-30/20/Micro***

Podręcznik użytkownika

GFK-0467L-PL

Czerwiec 1999

Znaki ostrzegawcze i uwagi

Niebezpieczeństwo

Symbolem tym oznaczono w niniejszym podręczniku informacje o niebezpiecznie wysokich napięciach, prądach o dużym natężeniu, temperaturach oraz innych czynnikach, związanych z urządzeniem lub współpracującym z nim sprzętem, które mogą spowodować obrażenia cielesne lub uszkodzenie urządzenia.

Ostrzeżenie

Symbolem tym oznaczono informacje, których nieprzestrzeganie może prowadzić do uszkodzenia urządzenia.

Uwaga

Symbolem tym oznaczono informacje o szczególnie dużym znaczeniu dla zrozumienia zasad eksploatacji i użytkowania urządzenia.

Niniejszy podręcznik przygotowano w oparciu o najnowsze, posiadane informacje. Podjęto wszelkie starania, aby zamieszczone informacje były dokładne, nie mniej jednak nie można zagwarantować, że uwzględnione zostały wszystkie szczegółowe dane i zmiany wprowadzone w sprzęcie i oprogramowaniu, jak również nie jest możliwe uwzględnienie wszystkich sytuacji, które mogą wystąpić w czasie instalowania, obsługi lub konserwacji urządzenia. Mogą występować różnice pomiędzy opisami zamieszczonymi w niniejszym dokumencie, a dostarczonymi urządzeniami i oprogramowaniem. GE Fanuc Automation nie zobowiązuje się do informowania właścicieli niniejszego podręcznika o wprowadzanych zmianach.

GE Fanuc Automation nie udziela żadnej gwarancji, jawnie sprecyzowanej lub domniemanej oraz nie ponosi odpowiedzialności za dokładność, kompletność, pełność oraz użyteczność informacji zawartych w niniejszym podręczniku. Nie jest udzielana gwarancja na przydatność handlową i techniczną.

Podane poniżej znaki towarowe są zastrzeżone przez GE Fanuc Automation North America, Inc.

Alarm Master	Genius	PROMACRO	Series Six
CIMPLICITY	Helpmate	PowerMotion	Series Three
CIMPLICITY 90-ADS	Logicmaster	PowerTRAC	VersaMax
CIMSTAR	Modelmaster	Series 90	VersaPro
Field Control	Motion Mate	Series Five	VuMaster
GENet	ProLoop	Series One	Workmaster

W niniejszym podręczniku opisano instrukcje oprogramowania Logicmaster90™ dla sterowników serii 90™-30 i 90-20 i 90-Micro. Sterowniki programowalne serii 90-30, 90-20 i 90 Micro wchodzi w skład większej rodziny sterowników serii 90 produkcji GE Fanuc Automation.

Zmiany uwzględnione w Podręczniku

- W jednostkach centralnych CPU364 (oprogramowanie systemowe w. 9.0 i nowsze) można przysyłać dane siecią Ethernet za pomocą jednego z dwóch, wbudowanych portów Ethernet. Jednostki te wyposażone są w port AAUI i port 10BaseT. Jest to jedyna jednostka centralna serii 90-30, obsługująca Dane globalne Ethernet (proszę porównać z informacjami na stronie 2-39).
- Nie można zmienić wskazań zegara czasu rzeczywistego za pomocą programatora, jeżeli włączona jest funkcja zabezpieczania pamięci przed zapisem za pomocą wyłącznika kluczowego (str. 2-14).
- Rozdział 2 uzupełniono o informacje odnośnie czasu wymaganego na obsługę modułu DSM.
- Szczegółowe informacje odnośnie sposobu zaimplementowania funkcji logarytmicznych/wykładniczych (str. 6-12) i funkcji TRUNC (str. 11-11) w jednostce centralnej CPU352.
- Zmodyfikowano opis bloku funkcyjnego SER, w celu lepszego zilustrowania jego możliwości. W załączniku podano szczegółowe informacje odnośnie czasu wykonywania tego bloku funkcyjnego. Blok funkcyjny SER jest obsługiwany przez jednostki centralne serii 35x i 36x, oprogramowanie systemowe w. 9.0 lub nowsza (str. 12-8).
- Wprowadzono nowy blok funkcyjny SVCREQ #45 (pominięcie obsługi wejść/wyjść w następnym cyklu), obsługiwany przez jednostki centralne serii 35x i 36x, oprogramowanie systemowe w. 9.0 lub nowsza.
- Skorygowano opis bloku parametrów dla funkcji SVCREQ #46 (str. 12-64).
- Wprowadzono inne zmiany i poprawki, stosownie do potrzeb.

Zawartość podręcznika

Rozdział 1. Wprowadzenie: Podstawowe informacje o rodzinie sterowników serii 90-30, serii 90-20 i Micro.

Rozdział 2. Działanie systemu: Opis działania sterowników serii 90-30/20/Micro. Omówiono cykl pracy sterownika, operacje wykonywane przez sterownik w czasie włączania i wyłączenia, zegary i przekaźniki czasowe, mechanizmy zabezpieczające przed nieuprawnionym dostępem, moduły wejść/wyjść oraz obsługę błędów działania. Zawarto również podstawowe informacje o programach sterujących w języku drabinkowym.

Rozdział 3. Błędy: Opis błędów dla sterowników serii 90-30, 90-20 i Micro. Omówiono również tabelę błędów działania sterownika, tabelę błędów działania układów wejść/wyjść oraz kategorie błędów.

Rozdziały 4 - 12: Opis instrukcji programowania sterowników serii 90-30/20/Micro. W każdym z tych rozdziałów omawiana jest grupa instrukcji.

Załącznik A. Czasy wykonywania elementów logicznych programu sterującego: Rozmiar pamięci w bajtach oraz czas wykonywania w mikrosekundach dla każdej z instrukcji programowania. Rozmiar pamięci to liczba bajtów, zajmowanych przez każdą z instrukcji w drabinie logicznej programu sterującego.

Załącznik B. Interpretacja tabeli błędów: Opis struktury komunikatów, odczytywanych z tabel błędów za pomocą oprogramowania 90-30/20/Micro.

Załącznik C. Nazwy mnemoniczne: W załączniku tym podano nazwy mnemoniczne, z których można korzystać w czasie przeszukiwania i edycji programu sterującego.

Załącznik D. Skróty klawiszowe: W załączniku tym zestawiono skróty klawiszowe wykorzystywane w oprogramowaniu LogicMaster 90-30/20/Micro. Zamieszczona w załączniku *wkładka* zawiera główne funkcje oraz ich nazwy mnemoniczne.

Załącznik E. Liczby zmiennoprzecinkowe: W załączniku tym omówiono różne zagadnienia, związane z realizacją operacji na liczbach zmiennoprzecinkowych.

Literatura uzupełniająca

Logicmaster™ 90 Series 90™-30/20/Micro Programming Software User's Manual (GFK-0466).

Logicmaster™ 90 Series 90-30 and 90-20 Important Product Information (GFK-0468).

Series 90™-30 Programmable Controller Installation Manual (GFK-0356).

Series 90™-20 Programmable Controller Installation Manual (GFK-0551).

Series 90™-30 I/O Module Specifications Manual (GFK-0898).

Series 90™ Programmable Coprocessor Module and Support Software User's Manual (GFK-0255).

Series 90™ PCM Development Software (PCOP) User's Manual (GFK-0487).

CIMPLICITY™ 90-ADS Alphanumeric Display System User's Manual (GFK-0499).

CIMPLICITY™ 90-ADS Alphanumeric Display System Reference Manual (GFK-0641).

Alphanumeric Display Coprocessor Module Data Sheet (GFK-0521).

Series 90™-30 and 90-20 PLC Hand-Held Programmer User's Manual (GFK-0402).

Power Mate APM for Series 90™-30 PLC—Standard Mode User's Manual (GFK-0840).

Power Mate APM for Series 90™-30 PLC—Follower Mode User's Manual (GFK-0781).

Motion Mate™ DSM302 for Series 90™-30 PLCs User's Manual (GFK-1464)

Series 90™-30 High Speed Counter User's Manual (GFK-0293).

Series 90™-30 Genius Communications Module User's Manual (GFK-0412).

Genius Communications Module Data Sheet (GFK-0272).

Series 90™-30 Genius™ Bus Controller User's Manual (GFK-1034).

Series 90™-70 FIP Bus Controller User's Manual (GFK-1038).

Series 90™-30 FIP Remote I/O Scanner User's Manual (GFK-1037).

Field Control™ Distributed I/O and Control System Genius™ Bus Interface Unit User's Manual (GFK-0825).

Series 90™ Micro Programmable Logic Controller User's Manual (GFK-1065).

Series 90™ PLC Serial Communications User's Manual (GFK-0582).

Rozdział 1	Wprowadzenie	1-1
Rozdział 2	Opis działania	2-1
	Część 1: Cykl pracy sterownika.....	2-2
	Standardowy cykl pracy sterownika.....	2-2
	Obliczanie czasu trwania cyklu pracy sterownika	2-7
	Komunikacja z programatorem	2-9
	Komunikacja systemowa (z innymi urządzeniami, modele 331 i wyższe)	2-11
	Komunikacja z modułem programowalnego koprocesora (modele 331 i wyższe) ..	2-12
	Komunikacja z modułem DSM	2-12
	Warianty standardowego cyklu pracy sterownika.....	2-13
	Tryb o stałym czasie trwania.....	2-13
	Cykl pracy sterownika w trybie STOP.....	2-13
	Tryby realizacji faz komunikacji	2-14
	Wyłącznik kluczowy w jednostkach centralnych serii 35x i 36x do zmiany trybu i zabezpieczenia przed zapisem.....	2-15
	Wyłącznik kluczowy w oprogramowaniu systemowym ver. 7 i nowsze.....	2-15
	Usuwanie zawartości tabeli błędów za pomocą wyłącznika kluczowego.....	2-16
	Dodatkowe zabezpieczenia pamięci w jednostkach centralnych z oprogramowaniem systemowym ver. 8 i nowszym	2-16
	Część 2: Organizacja programu, typy danych i typy zmiennych	2-17
	Bloki podprogramów (wyłącznie dla sterowników serii 90-30)	2-18
	Przykłady zastosowania podprogramów	2-18
	Wywoływanie podprogramów	2-19
	Procedury wykonywane okresowo	2-19
	Typy zmiennych	2-20
	Bity zmiany stanu i wymuszania wartości	2-21
	Pamięć stanu.....	2-21
	Typy danych	2-22
	Zmienne systemowe	2-23
	Podstawowe informacje o instrukcjach języka drabinkowego	2-26
	Format przekaźników i styków	2-26
	Format bloków funkcyjnych	2-26
	Parametry bloków funkcyjnych.....	2-28
	Przepływ sygnału przez blok funkcyjny	2-29
	Część 3: Rozruch i wyłączanie sterownika	2-30
	Rozruch sterownika	2-30
	Wyłączanie sterownika.....	2-33
	Część 4: Zegary	2-34
	Zegar odmierzający czas pracy sterownika	2-34
	Zegar czasu rzeczywistego	2-34
	Zegar wyłączający	2-35

Zegar czasu trwania przerwy w zasilaniu sterownika	2-35
Zegar cyklu pracy o stałym czasie trwania	2-35
Generator sygnału prostokątnego	2-36
Część 5: Mechanizmy zabezpieczające przed dostępem.....	2-37
Hasła	2-37
Zmiana poziomu uprawnień	2-38
Blokowanie/ odblokowywanie podprogramów	2-38
Trwała blokada podprogramów	2-38
Część 6: Moduły wejść/wyjść sterowników serii 90-30, 90-20 i Micro	2-39
Moduły wejść/wyjść sterowników serii 90-30	2-40
Formaty danych wejściowych/ wyjściowych	2-42
Domyślne wartości wyjść modułów serii 90-30	2-42
Dane diagnostyczne	2-42
Dane globalne	2-43
Dane globalne Genius	2-43
Dane globalne	2-43
Moduły wejść/wyjść sterowników serii 90-20	2-43
Konfigurowanie i programowanie	2-44
Rozdział 3 Błędy	3-1
Część 1: Obsługa błędów	3-2
Alarmy	3-2
Klasy błędów	3-2
Reakcja systemu na błędy	3-3
Tabele błędów działania	3-3
Wagi błędów	3-4
Zmienne do obsługi błędów	3-4
Wartości zmiennych do obsługi błędów	3-4
Dodatkowe skutki wystąpienia błędów działania	3-5
Tabela błędów działania sterownika	3-5
Tabela błędów działania układów wejść/wyjść	3-5
Wyświetlanie dodatkowych informacji o błędzie	3-6
Część 2: Tabela błędów działania sterownika	3-7
Wagi błędów	3-8
Loss of, or Missing, Option Module (Brak lub uszkodzenie modułu dodatkowego)	3-8
Reset of, Addition of, or Extra, Option Module (Ponowne uruchomienie, dodanie, lub brak konfiguracji dla modułu)	3-8
System Configuration Mismatch (Błędna konfiguracja systemu)	3-9
Option Module Software Failure (Błąd w oprogramowaniu wysprężalowanego modułu dodatkowego)	3-10
Program Block Checksum Failure (Błędna suma kontrolna programu)	3-10
Low Battery Signal (Rozładowanie baterii)	3-10
Constant Sweep Time Exceeded (Przekroczony czas trwania cyklu sterownika)	3-11
Application Fault (Błąd w działaniu programu sterującego)	3-11

	No User Program Present (Brak programu sterującego).....	3-12
	Corrupted User Program on Power-Up (Wykrycie uszkodzenia programu sterującego podczas rozruchu)	3-12
	Password Access Failure (Niepoprawne hasło dostępu sterownika)	3-12
	PLC CPU System Software Failure (Błąd w oprogramowaniu jednostki centralnej sterownika)	3-13
	Communications Failure During Store (Awaria komunikacji podczas ładowania programu do sterownika)	3-15
	Część 3: Tabela błędów działania układów wejść/wyjść	3-16
	Loss of I/O Module (Brak zadeklarowanego modułu wejść/wyjść)	3-16
	Addition of I/O Module (Dołączony niezadeklarowany moduł wejść/wyjść)	3-17
Rozdział 4	Styki i przekaźniki	4-1
	Styki.....	4-1
	Przekaźniki	4-2
	Styk otwarty— —	4-3
	Styk zamknięty — / —.....	4-3
	Przekaźnik o stykach otwartych —()—.....	4-3
	Przykład	4-3
	Przekaźnik o stykach zamkniętych —(/)—	4-4
	Przykład	4-4
	Przekaźnik o stykach otwartych z pamięcią —(M)—	4-4
	Przekaźnik o stykach zamkniętych z pamięcią —(/M)—	4-4
	Przekaźnik uaktywniany zboczem narastającym sygnału —(↑)—	4-4
	Przekaźnik uaktywniany zboczem opadającym sygnału —(↓)—	4-5
	Przykład	4-5
	Przekaźnik ustawialny SET —(S) —	4-5
	Przekaźnik ustawialny RESET —(R)—.....	4-5
	Przykład	4-6
	Przekaźnik SET z pamięcią —(SM)—	4-6
	Przekaźnik RESET z pamięcią —(RM)—	4-6
	Szyny	4-7
	Przykład	4-7
	Przekaźniki i styki kontynuacji	4-8
Rozdział 5	Liczniki i przekaźniki czasowe.....	5-1
	Blok danych sterujących przekaźnikami czasowymi i licznikami	5-1
	Przekaźnik czasowy załączający z pamięcią (ONDTR).....	5-3
	Parametry	5-4
	Dopuszczalne typy parametrów	5-4
	Przykład	5-5
	Przekaźnik czasowy załączający (TMR).....	5-5
	Parametry	5-6
	Dopuszczalne typy parametrów	5-6
	Przykład	5-7

Przełącznik czasowy wyłączający (OFDT).....	5-8
Parametry	5-9
Dopuszczalne typy parametrów	5-10
Przykład	5-10
Licznik zliczający w górę (UPCTR)	5-11
Parametry	5-11
Dopuszczalne typy parametrów	5-12
Przykład	5-12
Licznik zliczający w dół (DNCTR).....	5-12
Parametry	5-13
Dopuszczalne typy parametrów	5-13
Przykład	5-13
Przykład	5-14
Rozdział 6	
Funkcje matematyczne	6-1
Standardowe funkcje matematyczne	6-2
Parametry	6-3
Dopuszczalne typy parametrów	6-3
Przykład	6-3
Funkcje matematyczne a typy danych	6-4
Przykład	6-5
Dzielenie modulo (MOD).....	6-6
Parametry	6-6
Dopuszczalne typy parametrów	6-7
Przykład	6-7
Pierwiastek kwadratowy (SQRT) (INT, DINT, REAL).....	6-8
Parametry	6-8
Dopuszczalne typy parametrów	6-9
Przykład	6-9
Funkcje trygonometryczne (SIN, COS, TAN, ASIN, ACOS, ATAN)	6-10
Parametry	6-11
Dopuszczalne typy parametrów	6-11
Przykład	6-11
Funkcje logarytmiczne/ wykładnicze (LOG, LN, EXP, EXPT).....	6-12
Parametry	6-12
Dopuszczalne typy parametrów	6-13
Przykład	6-13
Konwersja miar kąta (RAD, DEG).....	6-14
Parametry	6-14
Dopuszczalne typy parametrów	6-14
Przykład	6-15

Rozdział 7	Funkcje relacji.....	7-1
	Standardowe funkcje relacji (EQ, NE, GT, GE, LT, LE).....	7-2
	Parametry	7-2
	Rozszerzony opis	7-3
	Dopuszczalne typy parametrów	7-3
	Przykład	7-3
	Zakres (INT, DINT, WORD)	7-4
	Parametry	7-5
	Dopuszczalne typy parametrów	7-5
	Przykład 1	7-5
	Przykład 2	7-6
Rozdział 8	Funkcje do operacji na ciągach bitów.....	8-1
	AND i OR (WORD).....	8-3
	Parametry	8-3
	Dopuszczalne typy parametrów	8-4
	Przykład	8-4
	XOR (WORD).....	8-5
	Parametry	8-5
	Dopuszczalne typy parametrów	8-6
	Przykład	8-6
	NOT (WORD).....	8-7
	Parametry	8-7
	Dopuszczalne typy parametrów	8-7
	Przykład	8-7
	SHL i SHR (WORD).....	8-8
	Parametry	8-9
	Dopuszczalne typy parametrów	8-9
	Przykład	8-9
	ROL i ROR (Word).....	8-10
	Parametry	8-10
	Dopuszczalne typy parametrów	8-11
	Przykład	8-11
	BTST (WORD).....	8-12
	Parametry	8-12
	Dopuszczalne typy parametrów	8-13
	Przykład	8-13
	BSET i BCLR (WORD).....	8-14
	Parametry	8-14
	Dopuszczalne typy parametrów	8-15
	Przykład	8-15
	BPOS (WORD).....	8-16
	Parametry	8-16
	Dopuszczalne typy parametrów	8-17
	Przykład	8-17

	MSKCMP (WORD, DWORD)	8-18
	Parametry	8-19
	Dopuszczalne typy parametrów	8-19
	Przykład	8-20
Rozdział 9	Funkcje do przesyłania danych	9-1
	MOVE (BIT, INT, WORD, REAL)	9-2
	Parametry	9-3
	Przykład 1	9-4
	Przykład 2	9-4
	BLKMOV (INT, WORD, REAL)	9-5
	Parametry	9-5
	Dopuszczalne typy parametrów	9-6
	Przykład	9-6
	BLKCLR (WORD).....	9-7
	Parametry	9-7
	Dopuszczalne typy parametrów	9-7
	Przykład	9-7
	SHFR (BIT, WORD).....	9-8
	Parametry	9-9
	Dopuszczalne typy parametrów	9-9
	Przykład 1	9-10
	Przykład 2	9-10
	BITSEQ (BIT).....	9-11
	Blok danych sterujących	9-11
	Parametry	9-12
	Dopuszczalne typy parametrów	9-13
	Przykład	9-13
	COMMREQ.....	9-14
	Blok danych sterujących	9-14
	Parametry	9-15
	Dopuszczalne typy parametrów	9-15
	Przykład	9-16
Rozdział 10	Funkcje do operacji tablicowych	10-1
	ARRAY_MOVE (INT, DINT, BIT, BYTE, WORD).....	10-2
	Parametry	10-3
	Dopuszczalne typy parametrów	10-3
	Przykład 1	10-4
	Przykład 2	10-4
	Przykład 3	10-5

Funkcje do przeszukiwania	10-6
Parametry	10-7
Dopuszczalne typy parametrów	10-7
Przykład 1	10-7
Przykład 2	10-8
Rozdział 11 Funkcje konwersji.....	11-1
—>BCD-4 (INT)	11-2
Parametry	11-2
Dopuszczalne typy parametrów	11-2
Przykład	11-2
—>INT (BCD-4, REAL).....	11-3
Parametry	11-3
Dopuszczalne typy parametrów	11-3
Przykład	11-4
—>DINT (REAL)	11-5
Parametry	11-5
Dopuszczalne typy parametrów	11-5
Przykład	11-6
—>REAL (INT, DINT, BCD-4, WORD).....	11-7
Parametry	11-7
Dopuszczalne typy parametrów	11-7
Przykład	11-8
—>WORD (REAL).....	11-9
Parametry	11-9
Dopuszczalne typy parametrów	11-9
Przykład	11-10
TRUN (INT, DINT)	11-11
Parametry	11-11
Dopuszczalne typy parametrów	11-11
Przykład	11-12
Rozdział 12 Funkcje sterujące	12-1
CALL.....	12-2
Przykład	12-2
DOIO	12-3
Parametry	12-4
Dopuszczalne typy parametrów	12-4
Uaktualnianie stanu wejść, przykład 1	12-5
Uaktualnianie stanu wejść, przykład 2	12-5
Uaktualnianie stanu wyjść, przykład 1	12-6
Uaktualnianie stanu wyjść, przykład 2.....	12-6
Rozszerzona funkcja DO I/O dla jednostek centralnych 331 i nowszych.....	12-7

SER.....	12-8
Charakterystyka	12-8
Przykład zastosowania bloku funkcyjnego SER.....	12-8
Parametry	12-9
Dopuszczalne typy parametrów	12-9
Blok danych sterujących	12-10
Słowo Status Extra Data.....	12-12
Blok danych	12-13
Opis działania.....	12-13
Tryby próbkowania	12-14
Przykłady	12-16
Przykład formatów zapisu czasu rejestrowania.....	12-20
END.....	12-21
Przykład	12-21
MCRN/MCR	12-22
Kompatybilność z jednostkami centralnymi	12-22
Opis działania funkcji MCRN.....	12-22
Opis działania funkcji MCR.....	12-23
Parametry	12-23
Różnice pomiędzy funkcjami MCR i JUMP.....	12-23
Przykład	12-24
ENDMCRN/ENDMCR	12-25
Przykład	12-25
JUMP	12-26
Przykłady	12-27
LABEL	12-28
Przykład	12-28
COMMENT.....	12-29
SVCREQ	12-30
Podstawowe informacje o bloku funkcyjnym SVCREQ.....	12-31
SVCREQ #1: Zmiana/ odczyt czasu trwania cyklu pracy sterownika w trybie o stałym czasie trwania	12-33
SVCREQ #2: Odczyt wartości z programatora.....	12-36
SVCREQ #3: Zmiana trybu i czasu komunikacji z programatorem	12-38
SVCREQ #4: Zmiana trybu i czasu trwania komunikacji systemowej.....	12-40
SVCREQ #6: Odczyt/ zmiana liczby słów sumy kontrolnej programu sterującego	12-42
SVCREQ #7: Odczyt/ zmiana wskazań zegara czasu rzeczywistego	12-44
SVCREQ #8: Zerowanie zegara wyłączającego	12-48
SVCREQ #9: Odczyt czasu trwania cyklu.....	12-49
SVCREQ #10: Odczyt nazwy folderu	12-50
SVCREQ #11: Odczyt identyfikatora sterownika	12-51
SVCREQ #12: Odczyt trybu pracy sterownika	12-52
SVCREQ #13: Zatrzymanie sterownika	12-53
SVCREQ #14: Kasowanie komunikatów z tablicy błędów działania sterownika i układów wejść/wyjść.....	12-54

	SVCREQ #15: Odczyt ostatnio zarejestrowanego komunikatu o błędzie działania	12-55
	SVCREQ #16: Odczyt wskazań zegara odmierzającego czas pracy sterownika	12-59
	SVCREQ #18: Kontrola występowania wymuszeń zmiany wartości zmiennych wejściowych i wyjściowych	12-60
	SVCREQ #23: Odczyt sumy kontrolnej programu sterującego i konfiguracji	12-61
	SVCREQ #26/30: Porównanie rzeczywistej konfiguracji modułów wejść/wyjść sterownika z zapisaną w pamięci	12-62
	SVCREQ #29: Zegar czasu trwania przerwy w zasilaniu sterownika	12-63
	SVCREQ #45: Pominięcie obsługi wejść/wyjść w następnym cyklu	12-64
	SVCREQ #46: Szybki dostęp do statusu	12-65
	PID	12-71
	Parametry	12-72
	Dopuszczalne typy parametrów	12-72
	Blok parametrów	12-73
	Opis działania bloku funkcyjnego PID	12-75
Załącznik A	Czasy wykonywania instrukcji	A-1
	Zapotrzebowanie instrukcji na pamięć w zaawansowanych modelach jednostek centralnych	A-11
	Czasy wykonywania dla przekaźników i styków	A-11
Załącznik B	Interpretacja tabeli błędów	B-1
	Tabela błędów działania sterownika	B-1
	Tabela błędów działania układów wejść/wyjść	B-8
Załącznik C	Nazwy skrótowe instrukcji	C-1
Załącznik D	Skróty klawiaturowe	D-1
Załącznik E	Liczby zmiennoprzecinkowe	E-1
	Informacje ogólne	E-1
	Format wewnętrznej reprezentacji liczb zmiennoprzecinkowych	E-3
	Wartości liczb zmiennoprzecinkowych	E-4
	Wprowadzanie i wyświetlanie liczb zmiennoprzecinkowych	E-5
	Błędy w operacjach i liczbach zmiennoprzecinkowych	E-6

Rysunek 2-1.	Cykl pracy sterownika	2-3
Rysunek 2-2.	Schemat blokowy komunikacji z programatorem	2-10
Rysunek 2-3.	Schemat blokowy komunikacji systemowej	2-11
Rysunek 2-4.	Komunikacja modułu programowalnego procesora (PCM) ze sterownikiem	2-12
Rysunek 2-5.	Sekwencja rozruchu	2-31
Rysunek 2-6.	Schemat pracy generatora sygnału prostokątnego	2-36
Rysunek 2-7.	Struktura wejść/wyjść sterowników serii 90-30	2-39
Rysunek 12-1.	Przykład próbkowania w trybie Rejestrowanie próbek do sygnału wyzwolenia	12-15
Rysunek 12-2.	Przykład próbkowania w trybie Rejestrowanie próbek do sygnału wyzwolenia	12-15
Rysunek 12-3.	Przykład próbkowania w trybie rejestrowania do zebrania określonej liczby próbek	12-16
Rysunek 12-4.	Regulator PID o niezależnych wyrazach (PIDIND)	12-80

Tabela 2-1.	Fazy cyklu pracy sterownika	2-4
Tabela 2-2.	Czasy odczytywania wejść i ustawiania wyjść dla sterowników 90-30, jednostki centralne 35x i 36x (w milisekundach)	2-5
Tabela 2-3.	Czasy odczytywania wejść i ustawiania wyjść dla sterowników 90-30, jednostki centralne do modelu 341 (w milisekundach).....	2-6
Tabela 2-4.	Zmienne rejestrowe	2-20
Tabela 2-5.	Zmienne dyskretne	2-20
Tabela 2-5.	Zmienne dyskretne - kontynuacja	2-21
Tabela 2-6.	Typy danych.....	2-22
Tabela 2-7.	Zmienne systemowe	2-23
Tabela 2-7.	Zmienne systemowe - kontynuacja	2-24
Tabela 2-7.	Zmienne systemowe - kontynuacja	2-25
Tabela 2-8.	Moduły wejść/wyjść serii 90-30	2-40
Tabela 2-8.	Moduły wejść/wyjść serii 90-30 - kontynuacja	2-41
Tabela 2-8.	Moduły wejść/wyjść serii 90-30 - kontynuacja	2-42
Tabela 3-1.	Zestawienie błędów	3-3
Tabela 3-2.	Wagi błędów i działania podejmowane w momencie ich wystąpienia	3-4
Tabela 4-1.	Typy styków	4-1
Tabela 4-2.	Typy przekaźników	4-2
Tabela 12-1	Blok danych sterujących dla przykładowego bloku funkcyjnego SER ...	12-17
Tabela 12-2	Zawartość próbki dla przykładowego bloku funkcyjnego SER.....	12-19
Tabela 12-3	Blok danych dla przykładowego bloku funkcyjnego SER.....	12-19
Tabela 12-4	Zestawienie bloków funkcyjnych SVCREQ.....	12-30
Tabela 12-5	Wartości parametrów wyjściowych dla funkcji odczytu dodatkowych danych	12-66
Tabela 12-6	Wartości parametrów wyjściowych dla funkcji zapisu dodatkowych danych	12-67
Tabela 12-7	Wartości parametrów wyjściowych dla funkcji zapisu/ odczytu dodatkowych danych.....	12-68
Tabela 12-8	Zestawienie parametrów bloku funkcyjnego PID.....	12-73
Tabela 12-9	Szczegółowy opis parametrów bloku funkcyjnego PID	12-76
Tabela 12-9	Zestawienie parametrów bloku funkcyjnego PID (kontynuacja).....	12-77
Tabela 12-9	Zestawienie parametrów bloku funkcyjnego PID (kontynuacja).....	12-78
Tabela A-1	Czasy wykonywania instrukcji, modele standardowe.....	A-2
Tabela A-1	Czasy wykonywania instrukcji, modele standardowe - kontynuacja.....	A-3
Tabela A-1	Czasy wykonywania instrukcji, modele standardowe - kontynuacja.....	A-4
Tabela A-1	Czasy wykonywania instrukcji, modele standardowe - kontynuacja.....	A-5
Tabela A-2	Czasy wykonywania instrukcji, modele zaawansowane.....	A-6
Tabela A-2	Czasy wykonywania, modele zaawansowane - kontynuacja	A-7
Tabela A-2	Czasy wykonywania, modele zaawansowane - kontynuacja	A-8
Tabela A-2	Czasy wykonywania, modele zaawansowane - kontynuacja	A-9
Tabela A-3	Czas wykonywania bloku funkcyjnego SER	A-10
Tabela A-4.	Zapotrzebowanie instrukcji na pamięć w jednostkach centralnych 350-352, 360, 363 i 364.....	A-11

Tabela B-1	Grupy błędów działania sterownika:.....	B-4
Tabela B-2	Tabela wag błędów działania sterownika:	B-5
Tabela B-3	Kody błędów alarmowych dla grupy PLC CPU Software Faults.....	B-5
Tabela B-4	Kody błędów alarmowych dla sterownika.....	B-6
Tabela B-5	Informacje o błędzie działania sterownika - Wykryty niewłaściwy kod logiczny Opcode.....	B-7
Tabela B-6	Czas i data wystąpienia błędu działania sterownika	B-7
Tabela B-7	Długość bloku dodatkowych informacji o błędzie.....	B-9
Tabela B-8	Adres zmiennej /wyjść	B-9
Tabela B-9	Kodowanie typów pamięci.....	B-9
Tabela B-10	Grupy błędów działania układów wejść/wyjść	B-10
Tabela B-11	Wagi błędów działania układów wejść/wyjść.....	B-11
Tabela B-12	Blok szczegółowych informacji o błędach działania układów wejść/wyjść	B-11
Tabela B-13	Czas i data wystąpienia błędu działania układów wejść/wyjść	B-12
Tabela E-1	Przebieg sygnału przy operacjach zmiennoprzecinkowych.....	E-7

Sterowniki serii 90-30, 90-20 i Micro są częścią większej rodziny sterowników programowalnych serii 90™. Odznaczają się one łatwością instalowania i konfigurowania, oferują zaawansowane mechanizmy programowania oraz są kompatybilne ze sterownikami serii 90-70.

Sterowniki serii 90-20 są efektywną ekonomicznie platformą dla aplikacji o małej liczbie wejść/wyjść. Seria sterowników 90-20 to:

- Małe sterowniki odznaczające się łatwością instalowania i eksploatacji.
- Efektywna ekonomicznie rodzina kompatybilnych sterowników PLC.
- Łatwa integracja systemu poprzez standardowe rozwiązania sprzętowe i protokoły komunikacyjne.

Sterowniki serii 90 Micro są efektywną ekonomicznie platformą dla aplikacji o małej liczbie wejść/wyjść. Przy opracowywaniu sterowników serii Micro postawione były takie same cele, jak dla serii 90-30. Cechy charakterystyczne serii Micro to:

- Mikroprocesor, zasilacz oraz moduły wejść/wyjść w formie jednego urządzenia o małych wymiarach gabarytowych.
- Wyposażenie większości modeli w licznik impulsów o wysokiej częstotliwości.
- Łatwość konfigurowania z uwagi na zintegrowanie w jednym urządzeniu mikroprocesora, zasilacza i modułów wejść/wyjść.

W oprogramowaniu dla sterowników serii 90-30, przeznaczonym dla jednostek centralnych CPU341 i starszych stosowane jest specjalne rozwiązanie do zarządzania pamięcią i priorytetami wykonywania w mikroprocesorze 80188. Modele 35x i 36x sterowników serii 90-30 korzystają z mikroprocesora 80386EX. Sterowniki serii 90 korzystają z mikroprocesora H8. Mikroprocesory te pozwalają zarówno na wykonywanie programu jak i realizację innych zadań, takich jak: przeprowadzanie czynności diagnostycznych, skanowanie wejść/wyjść i przetwarzanie alarmów. Oprogramowanie systemowe zawiera również procedury do komunikacji z programatorem. Procedury te pozwalają na usuwanie i wczytywanie programów sterujących, odczytywanie informacji o statusie oraz zapewniają kontrolę nad sterownikami.

W sterownikach serii 90-30, programy sterujące napisane przez użytkownika (przeznaczone do sterowania procesem) obsługiwane są poprzez specjalizowany koprocesor (ISCP). Koprocesor ten zrealizowany jest sprzętowo w systemach opartych na jednostce centralnej CPU Model 313 i wyższych, natomiast w systemach opartych na jednostce centralnej 311 i w sterownikach Micro zrealizowany jest programowo. Mikroprocesor 80188 i koprocesor ISCP pracują równolegle, przy czym mikroprocesor odpowiedzialny jest za komunikację sterownika, a koprocesor wykonuje większość instrukcji. Nie mniej jednak mikroprocesor musi również wykonywać zawarte w programie sterującym bloki funkcyjne, odpowiadające za operacje inne niż logiczne.

Wystąpienie pewnych uszkodzeń lub okoliczności mających wpływ na pracę i charakterystyki robocze systemu sygnalizowane jest w sterownikach serii 90-30, serii 90-20 i Micro w postaci błędów. Błędy te mogą uniemożliwić sterowanie maszyną czy procesem. Można też wyróżnić błędy mające na celu wyłącznie zwrócenie uwagi użytkownika, jak na przykład informacja o niskim stanie naładowania akumulatorów podtrzymujących zawartość pamięci i związanej z tym konieczności ich wymiany.

Błędy te rejestrowane są w dwóch tablicach (osobna dla sterownika i osobna dla układów wejść/wyjść). W jednostce centralnej 331 i wyższych, zapisywany jest również czas wystąpienia błędu. Tabele te można wyświetlić na ekranie komputera z pakietem do programowania.

Uwaga

Operacje zmiennoprzecinkowe mogą być realizowane **wyłącznie** w jednostkach centralnych 35x i 36x z oprogramowaniem systemowym ver. 9 lub nowsza oraz we wszystkich wersjach jednostek centralnych CPU352.

Spośród jednostek centralnych serii 90-30, tylko jednostka centralna CPU364 (oprogramowanie systemowe ver. 9.10 lub nowsza) obsługuje Dane globalne Ethernet.

Uwaga

Uzupełnieniem niniejszego podręcznika są załączniki, zawierające dodatkowe informacje.

- W Załączniku A podano rozmiar pamięci w bajtach oraz czas wykonywania w mikrosekundach każdej z instrukcji programowania.
- W Załączniku B opisano format reprezentacji komunikatów o błędach, zapisywanych w tabeli błędów sterownika i tabeli błędów układów wejść/wyjść.
- W załączniku C podano nazwy mnemoniczne instrukcji, wykorzystywane przy edycji programu sterującego.
- W załączniku D podano skróty klawiaturowe wykorzystywane w oprogramowaniu Logicmaster 90-30/20/Micro.
- W załączniku E omówiono zagadnienia związane z operacjami na liczbach zmiennoprzecinkowych.

W rozdziale tym opisano wybrane zagadnienia, związane z działaniem sterowników serii 90-30, 90-20 i Micro. Omówiono między innymi:

- Cykl pracy sterownika2-2
- Organizację programu sterującego, typy danych i typy zmiennych2-17
- Działania wykonywane przez sterownik podczas włączania i wyłączenia zasilania2-30
- Zegary.....2-33
- Mechanizm zabezpieczenia za pomocą haseł2-37
- Moduły wejść/wyjść sterowników serii 90-302-39

Część 1: Cykl pracy sterownika

W sterownikach serii 90-30, 90-20 i Micro, program sterujący wykonywany jest cyklicznie, aż do momentu zatrzymania za pośrednictwem instrukcji z komputera - programatora lub z innego urządzenia zewnętrznego. Ciąg operacji koniecznych do jednorazowego wykonania programu sterującego jest nazywany cyklem pracy sterownika. Oprócz wykonywania programu sterującego, cykl pracy sterownika zawiera również fazy gromadzenia danych z urządzeń wejściowych, wysłania informacji do urządzeń wyjściowych, przeprowadzenia wewnętrznej inicjalizacji sterownika, obsługi programatora oraz komunikacji z innymi urządzeniami.

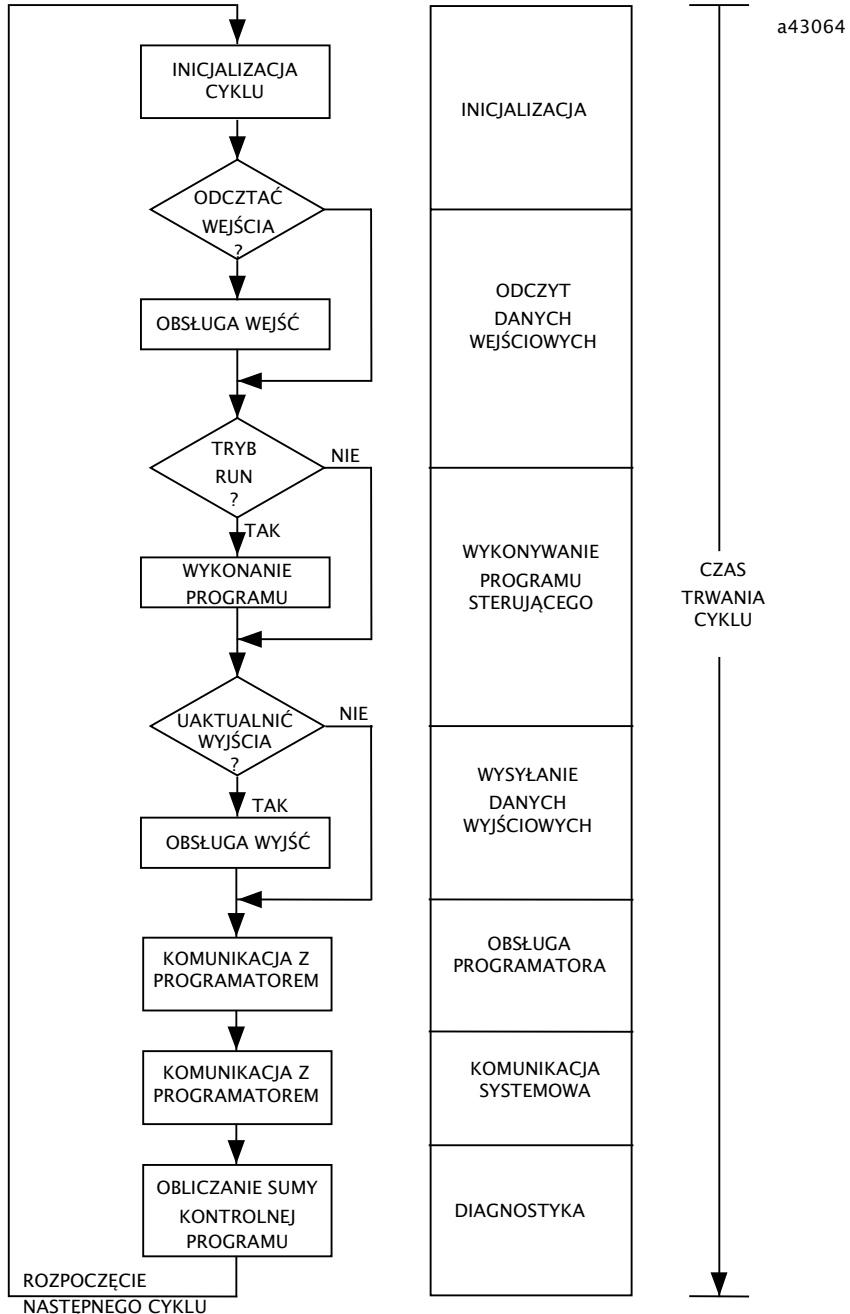
Sterowniki serii 90-30, 90-20 i Micro zwykle pracują w trybie standardowym (**STANDARD PROGRAM SWEEP**). Inne możliwe tryby pracy to tryb zatrzymania sterownika bez obsługi wejść/wyjść (**STOP WITH I/O DISABLED**), tryb zatrzymania sterownika z obsługą wejść/wyjść (**STOP WITH I/O ENABLED**) oraz tryb o stałym czasie trwania (**CONSTANT SWEEP**). Każdy z tych trybów jest uaktywniany przez parametry konfiguracyjne i wystąpienie pewnych zdarzeń w trakcie pracy systemu. Tryb pracy jest ustalany na początku każdego cyklu.

Standardowy cykl pracy sterownika

Jest to najczęściej wykorzystywany tryb pracy sterownika. W czasie pracy, jednostka centralna wykonuje program sterujący, obsługuje wejścia/ wyjścia, realizuje komunikację, itp. Czynności te powtarzane są cyklicznie. Standardowy tryb pracy składa się z siedmiu faz:

1. Inicjalizacji cyklu
2. Obsługi wejść
3. Wykonania programu sterującego
4. Obsługi wyjść
5. Obsługi programatora
6. Obsługi innych urządzeń
7. Diagnostyki

Wszystkie te fazy wykonywane są podczas każdego cyklu. Komunikacja z programatorem realizowana jest jedynie w przypadku wykrycia błędu działania lub zgłoszenia żądania komunikacji przez programator. Na zamieszczonym poniżej rysunku pokazano sekwencje faz standardowego cyklu pracy sterownika.



Rysunek 2-1. Cykl pracy sterownika

Zgodnie z powyższym schematem, każdy z cykli składa się z kilku faz. Czasy trwania tych faz podano w zamieszczonej poniżej tabeli.

Tabela 2-1. Fazy cyklu pracy sterownika

Faza cyklu	Opis	Czas trwania (ms) ⁴	
		Jednostki centralne 351, 352 i 36x (czasy dla jednostek centralnych serii 350 i 36x są takie same)	
Inicjalizacja	<ul style="list-style-type: none"> • Obliczanie czasu trwania cyklu • Zaplanowanie czasu rozpoczęcia następnego cyklu. • Określenie trybu pracy sterownika w następnym cyklu. • Uaktualnienie tabel błędów. • Wyzerowanie zegara wyłączającego. 	0.279	
Odczyt wejść	Odczyt danych wejściowych, uzyskanych z modułów wejść oraz z modułów opcjonalnych.	Proszę porównać z tabelą 2-2.	
Wykonanie programu sterującego	Wykonywanie programu sterującego przebiegiem procesu.	Czas wykonania programu zależy od długości programu i typów instrukcji stosowanych w programie. Czasy wykonania poszczególnych instrukcji podano w Załączniku A.	
Wysyłanie danych wyjściowych	Dane wyjściowe są wysyłane do modułów wyjść oraz do modułów opcjonalnych.	Proszę porównać z tabelą 2-2.	
Obsługa programatora	Przetwarzanie żądań nadesłanych przez programator oraz wyspecjalizowane moduły. ¹	Prog. ręczny	0.334
		LM-90	0.517
		PCM ²	0.482
Ponowna konfiguracja	Monitorowanie gniazd z uszkodzonymi modułami oraz gniazd bez modułów.	0.319 ⁶	
Diagnostyka	Weryfikacja integralności programu sterującego	0.010 dla każdego słowa sumy kontrolnej obliczanej w każdym cyklu ^{3,7}	

1. Czas obsługi urządzeń zewnętrznych zależy od trybu realizacji fazy komunikacji. W trybie **LIMITED**, czas trwania obsługi może wynosić maksymalnie 8 milisekund dla jednostek centralnych 311, 313, 323 i 331 oraz 6 milisekund dla jednostek centralnych serii 340 i wyższych. w trybie **RUN-TO-COMPLETION**, czas trwania obsługi może wynosić maksymalnie 50 ms, w zależności od liczby jednocześnie wysłanych żądań.
2. Pomiary przeprowadzono przy zainstalowanym fizycznie module programowalnego koprocesora (PCM), ale bez jego konfigurowania oraz bez uruchomionej aplikacji na tym module PCM.
3. Liczba słów sumy kontrolnej dla każdego z cykli może być zmieniona za pomocą funkcji SVCREQ.
4. Pomiary przeprowadzono bez programu sterującego oraz przy domyślnych parametrach konfiguracyjnych. Sterowniki serii 90-30 były zamontowane w kasecie z 10-oma gniazdami, bez podłączonych dodatkowych kaset rozszerzających. Ponadto dla czasów podanych w tabeli założono, że nie jest uruchomiony żaden podprogram wykonywany okresowo.
5. Czas odczytu danych wejściowych w sterowniku Micro można obliczyć wg następującego wzoru: $0.365 \text{ ms (część stała)} + 0.036 \text{ ms (czas filtrowania)} \times (\text{sumaryczny czas trwania cyklu}) / 0.5 \text{ ms}$.
6. Ponieważ sterownik Micro dysponuje statycznym zestawem wejść i wyjść, nie jest wymagane jego ponowne konfigurowanie.
7. Program dla sterownika Micro przechowywany jest w pamięci trwałej, w związku z czym nie jest sprawdzana jego integralność.

Tabela 2-2. Czasy odczytywania wejść i ustawiania wyjść dla sterowników 90-30, jednostki centralne 35x i 36x (w milisekundach)

Typ modułu	Jednostki centralne 35x i 36x			
	Kaseta główna	Kaseta rozszerzająca	Zdalna kaseta rozszerzająca	
Moduł wejść dyskretnych, 8 punktów	.030	.055	.206	
Moduł wejść dyskretnych, 16 punktów	.030	.055	.206	
Moduł wejść dyskretnych, 32 punktów	.043	.073	.269	
Moduł wyjść dyskretnych, 8 punktów	.030	.053	.197	
Moduł wyjść dyskretnych, 16 punktów	.030	.053	.197	
Moduł wyjść dyskretnych, 32 punktów	.042	.070	.259	
Kombinowany moduł wejść/wyjść dyskretnych	.060	.112	.405	
Moduł wejść analogowych, 4 kanały	.075	.105	.396	
Moduł wyjść analogowych, 2 kanałów	.058	.114	.402	
Moduł wejść analogowych (prądowych lub napięciowych), 16 kanałów	.978	1.446	3.999	
Moduł wyjść analogowych, 8 kanałów	1.274	1.988	4.472	
Zintegrowany moduł wejść/wyjść analogowych	1.220	1.999	4.338	
Licznik impulsów o wysokiej częstotliwości	1.381	2.106	5.221	
Procesor wejść/wyjść	1.574	2.402	6.388	
Interfejs Ethernet (bez połączenia)	.038	.041	.053	
Moduł Power Mate APM (1 oś)	1.527	2.581	6.388	
Moduł Power Mate APM (2 osie)	1.807	2.864	7.805	
DSM 302 *	40 AI, 6 AQ	2.143	3.315	9.527
	50AI, 9 AQ	2.427	3.732	11.092
	64 AI, 12 AQ	2.864	4.317	13.138
GCM	bez urządzeń	.911	1.637	5.020
	8 urządzeń po 64 słowa	8.826	16.932	21.179
GCM+	bez urządzeń	.567	.866	1.830
	32 urządzenia po 64 słowa	1.714	2.514	5.783
GBC	bez urządzeń	.798	1.202	2.540
	32 urządzenia po 64 słowa	18.382	25.377	70.777
Koproc.PCM 311	bez konfiguracji lub bez zadania dla aplikacji	.476	brak	brak
	odczyt w jak najkrótszym czasie 128%R	.485	brak	brak
ADC (bez zadania)		.476	brak	brak
Interfejs dla układów CNC i sterowników 90-70 (Master)	bez urządzeń	.569	.865	1.932
	16 urządzeń po 64 punkty	4.948	7.003	19.908
Interfejs dla układów CNC i sterowników 90-70 (Slave)	32 punkty	.087	.146	.553
	64 punkty	.154	.213	.789

* W przypadku systemów, w których czas trwania komunikacji z modułem DSM może powodować zakłócenia w pracy sterowanej maszyny, można przysyłać dane do/ odczytywać dane z modułu DSM za pomocą bloków funkcyjnych DO I/O, Suspend I/O i SVC_REQ #46, dzięki czemu, nie wszystkie dane będą odbierane w każdym cyklu. Szczegółowe informacje podano w podręczniku *Motion Mate DSM302 for Series 90-30 PLCs User's Manual*, GFK1464.

Tabela 2-3. Czasy odczytywania wejść i ustawiania wyjść dla sterowników 90-30, jednostki centralne do modelu 341 (w milisekundach)

Typ modułu	Model jednostki centralnej							
	311/313	331			340/341			
		Kaseta główna	Kaseta rozszerzaj.	Kaseta zdalna	Kaseta główna	Kaseta rozszerzaj.	Kaseta zdalna	
Moduł wejść dyskretnych, 8 punkty	.076	.054	.095	.255	.048	.089	.249	
Moduł wejść dyskretnych, 16 punkty	.075	.055	.097	.257	.048	.091	.250	
Moduł wejść dyskretnych, 32 punkty	.094	.094	.126	.335	.073	.115	.321	
Moduł wyjść dyskretnych, 8 punktów	.084	.059	.097	.252	.053	.090	.246	
Moduł wyjść dyskretnych, 16 punktów	.083	.061	.097	.253	.054	.090	.248	
Moduł wyjść dyskretnych, 32 punktów	.109	.075	.129	.333	.079	.114	.320	
Kombinowany moduł wejść/wyjść, 8 punktów	.165	.141	.218	.529	.098	.176	.489	
Moduł wejść analogowych, 4 kanały	.151	.132	.183	.490	.117	.160	.462	
Moduł wyjść analogowych, 2 kanały	.161	.138	.182	.428	.099	.148	.392	
Licznik impulsów o wysokiej częstotliwości	2.070	2.190	2.868	5.587	1.580	2.175	4.897	
Moduł Power Mate APM (1 oś)	2.330	2.460	3.175	6.647	1.750	2.506	5.899	
Moduł Power Mate APM (2 osie)	3.181	3.647	4.497	9.303	2.154	3.097	7.729	
DSM 302	40 AI, 6 AQ	3.613	4.081	5.239	11.430	2.552	3.648	9.697
	50AI, 9 AQ	4.127	4.611	5.899	13.310	2.911	4.170	11.406
	64 AI, 12 AQ	4.715	5.276	6.759	15.747	3.354	4.840	13.615
GCM	bez urządzeń	.041	.054	.063	.128	.038	.048	.085
	8 urządzeń 64 punktowych	11.420	11.570	13.247	21.288	9.536	10.648	19.485
GCM+	bez urządzeń	.887	.967	1.164	1.920	.666	.901	1.626
	32 urządzeń 64 punktowych	4.120	6.250	8.529	21.352	5.043	7.146	20.052
Koproc.PCM 311	bez konfiguracji lub bez zadania dla aplikacji	brak	3.350	brak	brak	1.684	brak	brak
	odczyt w jak najkrótszym czasie 128%R	brak	4.900	brak	brak	2.052	brak	brak
ADC 311	brak	3.340	brak	brak	1.678	brak	brak	
Moduł wejść analogowych (prądowych lub napięciowych), 16 kanałów	1.370	1.450	1.937	4.186	1.092	1.570	3.796	
Interfejs dla układów CNC i sterowników 90-70 (Master)	bez urządzeń	1.910	2.030	1.169	1.925	.678	.904	1.628
	szesnaście urządzeń 64 punktowych	6.020	6.170	8.399	21.291	4.992	6.985	20.010
Interfejs dla układów CNC i sterowników 90-70 (Slave)	32 punkty	.206	.222	.289	.689	.146	.226	.636
	64 punkty	.331	.350	.409	1.009	.244	.321	.926

* W przypadku systemów, w których czas trwania komunikacji z modułem DSM może powodować zakłócenia w pracy sterowanej maszyny, można przesyłać dane do/ odczytywać dane z modułu DSM za pomocą bloków funkcyjnych DO I/O, Suspend I/O i SVC_REQ #46, dzięki czemu, nie wszystkie dane będą odbierane w każdym cyklu. Szczegółowe informacje podano w podręczniku *Motion Mate DSM302 for Series 90-30 PLCs User's Manual*, GFK1464.

Obliczanie czasu trwania cyklu pracy sterownika

W tabeli 2-1 wyszczególniono siedem faz, które składają się na czas trwania cyklu pracy sterownika. Czas ten zawiera składniki stałe (jak inicjalizacja cyklu i diagnostyka) oraz zmienne. Składniki zmienne zależą od konfiguracji modułów wejść/wyjść, rozmiarów programu sterującego oraz typu urządzenia programującego podłączonego do sterownika (może to być komputer-programator wyposażony w oprogramowanie Logicmaster 90 lub programator ręczny; sterownik może też pracować bez podłączonego urządzenia programującego).

Przykład obliczania czasu trwania cyklu pracy sterownika

W zamieszczonej poniżej tabeli podano przykład obliczania czasu trwania cyklu pracy dla sterownika serii 90-30 z jednostką centralną CPU 331.

Obliczenia przeprowadzono dla następujących modułów i instrukcji:

- Moduły wejść: pięć 16-o punktowych modułów wejść sterownika serii 90-30.
- Moduły wyjść: cztery 16-o punktowe moduły wyjść serii 90-30.
- Instrukcje programu sterującego: Program składający się z 1200 kroków i zawierający 700 instrukcji logicznych (typu LD, AND, OR, itp.), 300 przełączników (OUT, OUTM, itp.) oraz 200 funkcji matematycznych (ADD, SUB, itp.).

Inicjalizacja

Inicjalizacja cyklu sterownika polega na wykonaniu wszystkich operacji koniecznych do rozpoczęcia cyklu. Jeśli sterownik pracuje w trybie o stałym czasie trwania (**CONSTANT SWEEP**), rozpoczęcie nowego cyklu jest opóźniane do momentu, aż upłynie czas przewidziany na wykonanie poprzedniego cyklu. Jeśli wymagany czas upłynął, a poprzedni cykl nie zakończył się, wartość zmiennej systemowej %SA0002, przypisanej do styku OV_SWP, jest ustawiana na 1, poczym nowy cykl jest rozpoczynany bez opóźnienia. Następnie uaktualniane są wartości zmiennych przypisanych do generatorów sygnału prostokątnego (o podstawie czasowej 0.01, 0.1 i 1 sekunda), poprzez obliczenie różnicy pomiędzy czasem rozpoczęcia poprzedniego i bieżącego cyklu. W celu zachowania dokładności, faktyczny czas rozpoczęcia cyklu zapisywany jest z dokładnością do 100 mikrosekund. Każdy generator sygnału prostokątnego posiada parametr podający liczbę okresów o długości 100 mikrosekund, które upłynęły od momentu ostatniego inkrementowania wartości generatora sygnału prostokątnego.

Odczyt danych wejściowych

W czasie przeznaczonym na obsługę wejść dokonywane jest odczytywanie stanu wejść sterownika, ma to miejsce bezpośrednio przed wykonaniem programu sterującego. W fazie tej odczytywany jest stan wejść wszystkich modułów sterowników 90-30, a odczytane wartości zapisywane są w pamięci adresowanej poprzez zmienne %I (wejścia dyskretne) oraz %AI (wejścia analogowe). Wszystkie dane globalne otrzymane za pośrednictwem modułu komunikacyjnego GENIUS, modułu Enhanced Genius lub za pośrednictwem sterownika szyny Genius (Genius Bus Controller) zostają zapisane w pamięci adresowanej przez zmienne globalne %G.

Wejścia są odczytywane zgodnie z rosnącymi adresami zmiennych, najpierw z modułu komunikacyjnego GENIUS, następnie z modułów wejść dyskretnych i ostatecznie z modułów wejść analogowych.

Jeśli sterownik znajduje się w trybie zatrzymania bez obsługi wejść/wyjść (**STOP/NO IO**), opisana tu faza jest pomijana.

Wykonywanie programu sterującego

Faza ta rozpoczyna się natychmiast po zakończeniu odczytywania wejść. Program sterujący wykonywany jest zawsze w kolejności instrukcji, począwszy od pierwszej instrukcji pierwszego szeregu programu. Wykonanie programu sterującego powoduje zmianę stanu zmiennych przypisanych wyjściom sterownika. Program sterujący wykonywany jest do momentu napotkania instrukcji END (instrukcja END jest niewidoczna, chyba że wykorzystywany jest programator ręczny).

Program sterujący wykonywany jest przez mikroprocesor 80C188 oraz przez koprocesor ISCP. W przypadku sterowników wyposażonych w jednostki centralne CPU 313 i wyższe, koprocesor ISCP wykonuje operacje logiczne, a procesor 80C188 lub 80386EX wykonuje operacje odpowiadające blokom funkcyjnym, licznikom oraz przełącznikom czasowym. W przypadku sterowników wyposażonych w jednostki centralne CPU 311 i sterowników serii 90-20, wszystkie wymienione wyżej instrukcje wykonywane są przez procesor 80C188. W sterownikach Micro, procesor H8 wykonuje wszystkie operacje logiczne oraz bloki funkcyjne.

Czasy wykonywania poszczególnych instrukcji logicznych programu sterującego podano w Załączniku A.

Wysyłanie danych wejściowych

W fazie tej następuje przypisanie wartości zmiennych wyjściowych do fizycznych wyjść sterownika, etap ten realizowany jest natychmiast po zakończeniu wykonywania programu sterującego. Stan wyjść jest uaktualniany na podstawie wartości zmiennych %Q (wyjścia dyskretne) oraz %AQ (wyjścia analogowe). Jeśli moduł komunikacyjny GENIUS jest skonfigurowany do przesyłania danych globalnych, dane z komórek pamięci adresowanych przez zmienne globalne %G zostają przesłane do modułu GCM, GCM+ lub GBC. W sterownikach serii Micro obsługiwane są wyłącznie wyjścia dyskretne.

We wszystkich sterownikach serii 90-30 moduły wyjść obsługiwane są wg. rosnących adresów zmiennych.

Jeśli sterownik znajduje się w trybie zatrzymania bez obsługi wejść/wyjść (**STOP/NO IO**), opisana tu faza zostaje pominięta. Zakończenie fazy obsługi wyjść następuje w momencie, gdy wartości zmiennych przypisanych wyjściom zostaną przesłane do wszystkich modułów wyjść serii 90-30.

Obliczanie sumy kontrolnej programu sterującego

Na końcu każdego cyklu obliczana jest suma kontrolna programu sterującego użytkownika. Ponieważ obliczanie sumy kontrolnej dla całego programu jest bardzo czasochłonne, można określić liczbę słów z zakresu 0 do 32, dla których ma być obliczana suma kontrolna.

Jeżeli obliczona suma kontrolna nie jest zgodna z zapamiętaną, następuje ustawienie znacznika błędu. Powoduje to wprowadzenie nowej pozycji do tabeli błędów sterownika oraz przejście do trybu **STOP**. W przypadku błędu sumy kontrolnej, nie są wprowadzane żadne zmiany w fazie komunikacji z programatorem. Domyślnie, suma kontrolna jest obliczana dla 8 słów.

Komunikacja z programatorem

Ta faza cyklu przeznaczona jest na komunikację z programatorem. Faza ta jest wykonywana, jeżeli do sterownika podłączony jest programator, a w systemie znajduje się moduł wymagający konfiguracji. Jeśli żaden z tych warunków nie jest spełniony, faza komunikacji z programatorem jest pomijana. Podczas jednego cyklu może zostać skonfigurowany tylko jeden moduł.

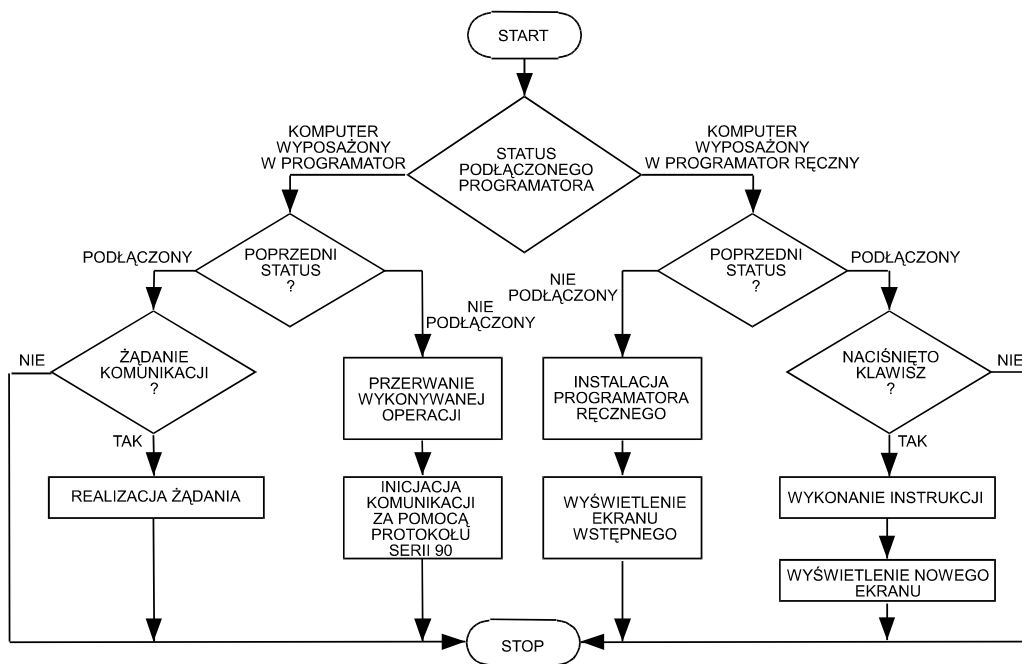
Sterownik umożliwia podłączenie programatora ręcznego lub innych programatorów, mogących komunikować się za pomocą złącza szeregowego i korzystających z protokołu SNP. Zapewniana jest również obsługa komunikacji programatora z wyspecjalizowanymi modułami dodatkowymi.

W domyślnym trybie komunikacji ograniczonej czasowo, jednostka centralna sterownika wykonuje jedną operację związaną z programatorem podczas każdego cyklu, co oznacza, że odpowiada na jedno żądanie komunikacji lub na naciśnięcie klawisza. Jeśli programator wysła żądanie komunikacji, którego realizacja zabiera więcej niż 6 milisekund (lub 8 w zależności od wersji jednostki centralnej - proszę porównać z uwagą poniżej), to jego realizacja zostaje podzielona na kilka cykli tak, aby żaden z nich nie trwał więcej niż maksymalnie 6 milisekund (lub 8 w zależności od wersji jednostki centralnej - proszę porównać z uwagą).

Uwaga

Maksymalny, dopuszczalny czas trwania komunikacji wynosi 6 milisekund dla jednostki centralnej 340 i wyższych oraz 8 milisekund dla jednostek centralnych 311, 313, 323 i 331.

Poniżej przedstawiono schemat blokowy fazy komunikacji z programatorem.

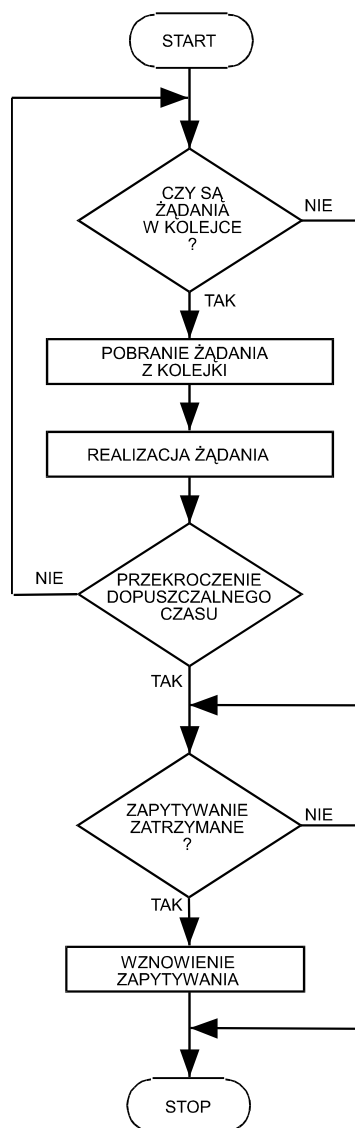


Rysunek 2-2. Schemat blokowy komunikacji z programatorem

Komunikacja systemowa (z innymi urządzeniami, modele 331 i wyższe)

W fazie komunikacji systemowej realizowane są żądania komunikacji nadsyłane przez wyspecjalizowane moduły, np. moduł programowalnego koprocesora lub DSM (proszę porównać ze schematem). Żądania komunikacji są obsługiwane w kolejności napływania. Ponieważ moduły dodatkowe są zapytywane metoda okreśną, żaden z tych modułów nie jest uprzywilejowany w stosunku do innych.

Jeżeli sterownik pracuje w trybie **RUN-TO-COMPLETION**, czas trwania fazy komunikacji systemowej ograniczony jest do 50 ms. Jeśli czas realizacji żądania komunikacji wysłanego przez moduł dodatkowy przekracza 50 milisekund, jest ono realizowane w kilku cyklach tak, że żaden cykl nie zostaje przedłużony o więcej niż 50 ms.

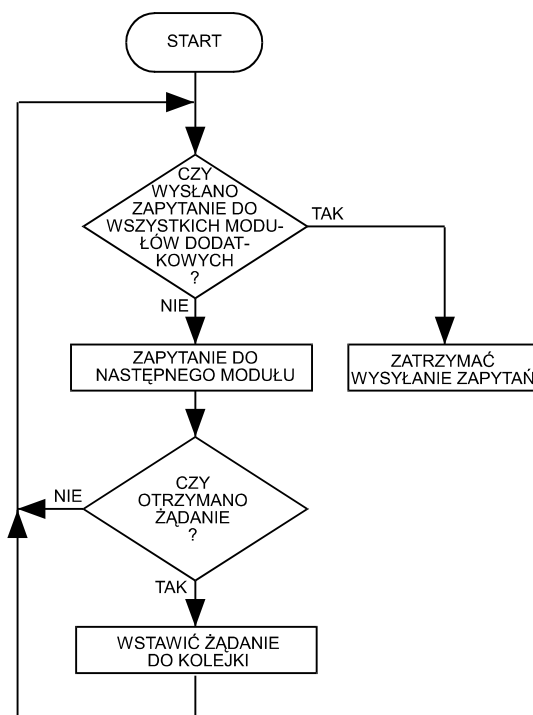


Rysunek 2-3. Schemat blokowy komunikacji systemowej

Komunikacja z modułem programowalnego koprocесora (modele 331 i wyższe)

Wyspecjalizowane moduły dodatkowe, jak moduł programowalnego koprocесora (PCM), nie są w stanie przerwać pracy jednostki centralnej w celu nawiązania komunikacji. Jednostka centralna musi sama sprawdzić, czy któryś z modułów nie żąda komunikacji. Sprawdzanie to może być realizowane asynchronicznie w czasie trwania wykonywania cyklu (proszę porównać z zamieszczonym poniżej schematem).

Jeżeli po wysłaniu zapytania do modułu dodatkowego, w odpowiedzi przesłane zostanie żądanie komunikacji, jest ono wstawiane do kolejki w czasie fazy komunikacji systemowej.



Rysunek 2-4. Komunikacja z modułem programowalnego koprocесora

Komunikacja z modułem DSM

Moduł DSM302 to wyspecjalizowany moduł pracujący asynchronicznie z jednostką centralną sterowników serii 90-30. Dane są przesyłane pomiędzy jednostką centralną a modułem DSM automatycznie.

W module DSM można skonfigurować trzy różne typy danych %AI i %AQ. Wymiana danych poprzez magistralę, pomiędzy jednostką centralną sterownika a modułem DSM wymaga pewnego czasu. W tabeli 2-2 podano typowe wartości czasu dla różnych konfiguracji danych %AI i %AQ. Szczegółowe informacje odnośnie modułu DSM302 podano w podręczniku *Motion Mate DSM302 for Series 90-30 PLCs User's Manual*, GFK-1464.

Warianty standardowego cyklu pracy sterownika

W standardowym cyklu pracy sterownika można wprowadzić lub wymusić pewne zmiany. Zmiany te, opisane poniżej, mogą być dokonane za pomocą oprogramowania.

Tryb o stałym czasie trwania

W standardowym trybie pracy, każdy cykl jest wykonywany tak szybko, jak to jest możliwe, co powoduje, że czasy trwania poszczególnych cykli może się różnić. Trybem alternatywnym jest tryb o stałym czasie trwania (**CONSTANT SWEEP TIME**), w którym każdy cykl trwa przez taki sam okres czasu. Tryb ten przyjmowany jest domyślnie po ustawieniu parametru Configured Constant Sweep, będzie on zawsze ustawiany po przejściu z trybu zatrzymania (**STOP**) do trybu pracy (**RUN**). Czas trwania tego cyklu może wynosić od 5 do 200 milisekund (lub do 500 milisekund dla sterowników serii 35x i 36x), domyślnie ustawiane jest 100 milisekund.

Z powodu zmian czasu trwania poszczególnych faz cyklu w różnych cyklach, czas ten powinien być co najmniej 10 ms wyższy od czasu trwania cyklu wyświetlanego w linii statusu sterownika podczas jego pracy w trybie standardowym (**NORMAL SWEEP**). Ma to na celu uniknięcie wystąpienia błędów działania sterownika, powodowanych przez przekroczenie dopuszczalnego czasu trwania.

Omawiany tryb pracy może być stosowany, jeżeli punkty wejść/wyjść lub wartości rejestrów muszą być sprawdzane ze stałą częstotliwością, co jest wymagane w wielu algorytmach sterowania. Jednym z powodów stosowania tego trybu pracy może być też konieczność zapewnienia ustawiania wyjść sterownika w stałych odstępach czasu. Innym, może być konieczność wprowadzenia pewnego odstępu czasowego pomiędzy fazą obsługi wyjść sterownika a fazą obsługi wejść w następnym cyklu, co ma na celu zapewnienie czasu do ustabilizowania stanu wejść po otrzymaniu danych wyjściowych z programu.

Jeśli czas przeznaczony na cykl upłynie przed zakończeniem wykonywania tego cyklu, cały cykl, włącznie z fazami komunikacji z programatorem i innymi urządzeniami, zostaje dokończony. Jednakże na początku następnego cyklu zarejestrowany zostanie błąd przekroczenia czasu trwania cyklu.

Uwaga

W odróżnieniu od dynamicznie ustawianego w trakcie pracy sterownika parametru Constant Sweep, zmiana parametru konfiguracyjnego Configured Constant Sweep może być dokonana wyłącznie w trybie **STOP** i wymagana ponownego przesłania konfiguracji w celu uwzględnienia wprowadzonych zmian. Po przesłaniu nowej konfiguracji, tryb ten będzie przyjmowany jako domyślny.

Cykl pracy sterownika w trybie STOP

W trybie **STOP** program sterujący nie jest wykonywany. Kontynuowana jest komunikacja z programatorem oraz z wyspecjalizowanymi modułami opcjonalnymi. Dodatkowo, w trybie **STOP** kontynuowany jest odczyt danych z modułów wejść/wyjść oraz ich kontrola pod kątem wykrycia ewentualnych błędów w działaniu.

W celu zwiększenia efektywności, system operacyjny przeznaczony na komunikację więcej czasu niż przy pracującym sterowniku (zwykle około 50 ms na każdą fazę komunikacji). Wprowadzając odpowiednią konfigurację, można określić czy mają być obsługiwane wejścia i wyjścia modułów. Wejścia i wyjścia modułów są obsługiwane w trybie **STOP** jeżeli parametr **IOScan-Stop** na ekranie do konfigurowania jednostki centralnej CPU zostanie ustawiony na wartość **YES**.

Tryby realizacji faz komunikacji

Domyślnie, czas przeznaczony na komunikację z programatorem jest ograniczony. Oznacza to, że jeśli czas realizacji żądania przekracza 6 milisekund, jest ono realizowane w kilku cyklach tak, że żaden cykl nie zostaje przedłużony o więcej niż 6 ms. Dla jednostek centralnych CPU 313, 323 i 331, w trybie programowania on-line podczas pracy (**RUN**), cykl może być przedłużony o maksymalnie 12 milisekund. Tryb realizacji faz komunikacji można zmienić na ekranie "Sweep Control" oprogramowania Logicmaster, zgodnie z instrukcjami podanymi w Rozdziale 5 podręcznika *Logicmaster 90™ Series 90™-30/20/Micro Programming Software User's Manual* (GFK-0466).

Uwaga

Jeżeli czas przeznaczony na komunikację z programatorem jest ograniczony, dodatkowe moduły takie jak moduł PCM i GBC, obsługiwane w fazie komunikacji systemowej, będą w mniejszym stopniu wpływać na czas realizacji cyklu, ale spowoduje to wolniejszą obsługę wysyłanych przez nie żądań.

Wyłącznik kluczowy w jednostkach centralnych serii 35x i 36x do zmiany trybu i zabezpieczenia przed zapisem

Każda jednostka centralna 35x i 36x posiada umieszczony z przodu modułu wyłącznik kluczowy, pozwalający na zabezpieczenie pamięci przed dokonaniem zmian. Po ustawieniu przełącznika w pozycji **ON/RUN**, zablokowana zostaje możliwość zmiany zawartości pamięci, pod warunkiem ustawienia odpowiedniego parametru konfiguracyjnego.

Począwszy od wersji 7 oprogramowania systemowego jednostek centralnych 351 i 352, wyłącznik ten posiada dodatkową funkcję: pozwala on na zatrzymanie sterownika (**STOP**), rozpoczęcie pracy (**RUN**) oraz usunięcie błędów o nie krytycznym znaczeniu, opisanych w następnym punkcie.

Począwszy od wersji 8 oprogramowania systemowego jednostek centralnych 35x i 36x, wyłącznik ten realizuje funkcje dodatkowe zabezpieczenia pamięci: zapewnia on dwa dodatkowe mechanizmy zabezpieczania pamięci (proszę porównać z punktem "Korzystanie z mechanizmów zabezpieczających pamięci w oprogramowaniu systemowym ver. 8 i nowsze).

Jeżeli aktywowana jest funkcja zabezpieczania pamięci przed dokonaniem zmian i wyłącznik jest ustawiony w pozycji **ON/RUN**, wskazania zegara czasu rzeczywistego można zmienić wyłącznie za pomocą pakietu do programowania. Nie można zmienić wskazań zegara czasu rzeczywistego za pomocą programatora, jeżeli włączona jest funkcja zabezpieczania pamięci przed dokonaniem zmian za pomocą wyłącznika kluczowego.

Wyłącznik kluczowy w oprogramowaniu systemowym ver. 7 i nowsza

W odróżnieniu od mechanizmów zabezpieczania pamięci dostępnych we wcześniejszych wersjach, jeżeli funkcja zabezpieczania pamięci przed zmianami nie zostanie włączona poprzez odpowiednie ustawienie parametru konfiguracyjnego jednostki centralnej **RUN/STOP Key Switch**, nie będą włączone omawiane tu mechanizmy zabezpieczające pamięć.

Mechanizm działania tego wyłącznika i realizowane przez niego czynności kontrolne są takie same, jak w przypadku normalnego rozruchu sterownika, tzn. sterownik nie może przejść do trybu **RUN**, jeżeli znajduje się w trybie **STOP/FAULT**. Nie mniej jednak, za pomocą tego wyłącznika można w trybie **STOP/FAULT** usunąć błędy o nie krytycznym znaczeniu oraz przełączyć sterownik za pomocą wyłącznika kluczowego do trybu pracy (**RUN**).

Jeżeli tabela błędów zawiera błędy o nie krytycznym znaczeniu (tzn. takie które nie powodują przejścia jednostki centralnej do trybu **STOP/FAULT**), jednostka centralna zostanie przełączona do trybu **RUN**, po pierwszym przestawieniu przełącznika z pozycji **STOP** do pozycji **RUN**, bez usuwania zawartości tabeli błędów.

Jeżeli w tabeli błędów zapisane są błędy **krytyczne** (jednostka centralna w trybie **STOP/FAULT**), pierwsze przestawienie przełącznika z pozycji **STOP** do pozycji **RUN** spowoduje błyskanie lampki sygnalizacyjnej z częstotliwością 2 Hz przez okres 5 sekund. Błyskanie lampki sygnalizacyjnej **RUN** informuje, że w tabeli błędów znajdują się błędy o krytycznym znaczeniu. W każdym z tych przypadków, ustawienie wyłącznika w pozycji **RUN** nie powoduje przejścia jednostki centralnej CPU do trybu **RUN**.

Usuwanie zawartości tabeli błędów za pomocą wyłącznika kluczowego

Jeżeli przełącznik zostanie przestawiony z pozycji **RUN** do pozycji **STOP**, a następnie ponownie do pozycji **RUN**, przed upływem 5 sekund, w czasie błyskania lampki sygnalizacyjnej **RUN**, zawartość tabeli błędów zostanie usunięta, a jednostka centralna przejdzie do trybu **RUN**. Spowoduje to zatrzymanie błyskania lampki sygnalizacyjnej i jej zapalenie. Przed ustawieniem w innej pozycji, przełącznik musi pozostawać w pozycji **RUN** lub **STOP** przez co najmniej 1/2 sekundy.

Uwaga

Po upływie 5 sekund (lampka **RUN** przestaje błyskać), jednostka centralna powróci do swojego pierwotnego stanu, trybu **STOP/ RUN**, a w tabeli będą nadal przechowywane błędy. Jeżeli wyłącznik kluczowy zostanie wtedy przestawiony z pozycji **STOP** ponownie do pozycji **RUN**, proces ten będzie powtórzony, tak jakby to było pierwsze przestawienie.

W zamieszczonej poniżej tabeli przedstawiono wpływ parametrów konfiguracyjnych jednostki centralnej (R/S Switch i IOScan-Stop) na działanie przełącznika oraz wpływ fizycznego ustawienia przełącznika na działanie sterownika.

Parametr konfiguracyjny CPU R/S Key Switch	Pozycja przełącznika	Parametr konfiguracyjny CPU IOScan-Stop	Działanie sterownika
OFF	X	X	Dostępne są wszystkie tryby programowania i pracy sterownika.
ON	ON/RUN	X	Dostępne są wszystkie tryby programatora sterownika.
ON	OFF/STOP	X	Sterownik nie może być przełączony do trybu pracy RUN .
ON	Przestawienie wyłącznika kluczowego z pozycji OFF/STOP do pozycji ON/RUN.	X	Sterownik zostaje przełączony do trybu pracy RUN , pod warunkiem że tabela błędów jest pusta, w przeciwnym wypadku dioda RUN będzie błyskać przez 5 sekund.
ON	Przestawienie wyłącznika kluczowego z pozycji ON/RUN do pozycji OFF/STOP.	NO	Sterownik zostaje przełączony do trybu STOP-bez obsługi wejść/wyjść.
ON	Przestawienie wyłącznika kluczowego z pozycji ON/RUN do pozycji OFF/STOP.	YES	Sterownik zostaje przełączony do trybu STOP-IO.

X=Nie ma żadnego wpływu, bez względu na zajmowaną pozycję.

Dodatkowe zabezpieczenia pamięci w jednostkach centralnych z oprogramowaniem systemowym wer. 8 i nowszym

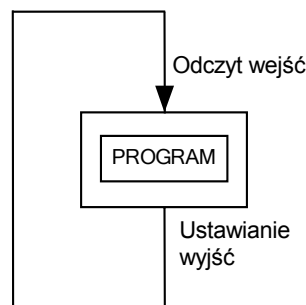
W jednostkach centralnych z oprogramowaniem systemowym wer. 8 i nowszym, wyłącznik kluczowy posiada całą funkcjonalność przedstawioną powyżej oraz dodatkowo, po ustawieniu odpowiedniego parametru konfiguracyjnego, pozwala zabezpieczyć pamięć RAM przed wprowadzaniem zmian z poziomu oprogramowania. Po aktywowaniu tego mechanizmu zabezpieczającego, następuje zablokowanie dwóch rodzajów działań: możliwości zmiany programu sterującego użytkownika i konfiguracji oraz możliwości ustawiania i zmiany wartości przechowywanych w pamięci parametrów (wymuszenia stanów sygnałów). Mechanizm ten włączany jest z poziomu oprogramowania Logicmaster, poprzez ustawienie pola Mem Protect w konfiguracji jednostki centralnej serii 35x lub 36x. Domyślnie ustawiona jest wartość Disabled.

Część 2: Organizacja programu, typy danych i typy zmiennych

Zamieszczona poniżej tabela zawiera maksymalną, dopuszczalną wielkość programów sterujących dla sterowników serii 90-30.

Modele	Wielkość programu sterującego (Kb)
CPU311	6
CPU313, CPU323	12
CPU331	16
CPU340	32
CPU341	80
CPU350	80 (wer. 9 i nowsze) 32 (przed wersją 9)
CPU351, CPU352, CPU360, CPU363, CPU364	240 (wer. 9 i nowsze) 80 (przed wersją 9)

Począwszy od jednostek centralnych z oprogramowaniem systemowym wer. 9, niektóre wielkości pamięci dla serii 351, 352 i 36x mogą być konfigurowane. (Szczegółowe instrukcje oraz dostępne rozmiary pamięci omówiono w podręczniku *Logicmaster 90™ Series 90™-30/20/Micro Programming Software User's Manual (GFK-0466K lub wersje nowsze)*. Wielkość programu sterującego dla sterowników serii 90-20 z jednostką centralną CPU 211 może wynosić do 2 Kb. Program użytkownika zawiera część logiczną, wykonywaną w momencie jego uruchomienia. Jeden blok logiczny (blok główny lub podprogram) może zawierać maksymalnie 3000 szczebli W przypadku sterowników serii 90-30, maksymalny rozmiar bloku C wynosi 80 KB a bloków języka drabinkowego i bloku SFC do 16 KB, należy jednak pamiętać, że w niektórych blokach SFC część z tych 16 KB przeznaczana jest na blok danych wewnętrznych. Część logiczna programu wykonywana jest przez sterownik cyklicznie.



Rozmiary programów oraz limity adresów zmiennych poszczególnych typów dla każdej z jednostek centralnych podano w podręczniku *Series 90-30 Programmable Controller User's Manual, GFK-0356* lub w podręczniku *Series 90-20 Programmable Controller User's Manual, GFK-0551*.

Wszystkie programy rozpoczynają się tabelą deklaracji zmiennych, w której wyszczególnione są nazwy pomocnicze i objaśnienia zmiennych.

Edytor deklaracji bloków programu podaje wykaz podprogramów zadeklarowanych w programie.

Bloki podprogramów (wyłącznie dla sterowników serii 90-30)

Z poziomu wykonywanego aktualnie programu można wywoływać podprogramy. Podprogram musi zostać zadeklarowany przy pomocy edytora deklaracji bloków programu; dopiero wtedy podprogram ten można wywołać za pomocą instrukcji CALL. Maksymalnie w programie można zadeklarować 64 podprogramy, a każdy z bloków programu sterującego może zawierać do 64 instrukcji CALL. Maksymalny rozmiar podprogramów to 16 Kb lub 3000 szczebli, ale należy pamiętać, że program główny wraz ze wszystkimi podprogramami musi zmieścić się w granicach obowiązujących dla poszczególnych jednostek centralnych.

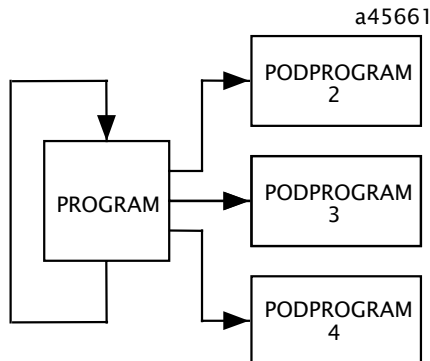
Uwaga

Nie można korzystać z podprogramów w sterownikach serii 90-20 i Micro.

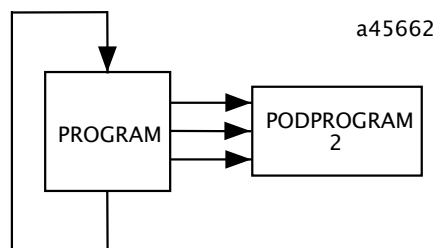
Używanie podprogramów jest opcjonalne. Podzielenie jednak programu na mniejsze bloki może uprościć proces programowania, ułatwić zrozumienie algorytmu sterowania oraz zmniejszyć globalną liczbę elementów logicznych w programie.

Przykłady zastosowania podprogramów

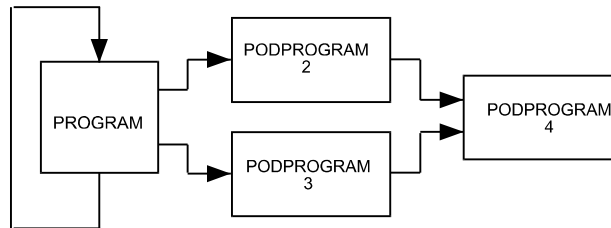
Przykładem wykorzystania podprogramów może być podział części logicznej programu na trzy podprogramy, z których każdy może być wywoływany w razie potrzeby z programu głównego. Główny blok programu sterującego może wtedy zawierać małą liczbę instrukcji logicznych, służących głównie do obsługi podprogramów.



Podprogram może być wywoływany dowolną liczbą razy podczas wykonywania programu sterującego. Bloki logiczne, które powtarzają się w programie sterującym, mogą być zatem wprowadzone w postaci podprogramów. Można je następnie wywoływać w odpowiednich miejscach programu. Pozwala to na zmniejszenie całkowitych rozmiarów programu.



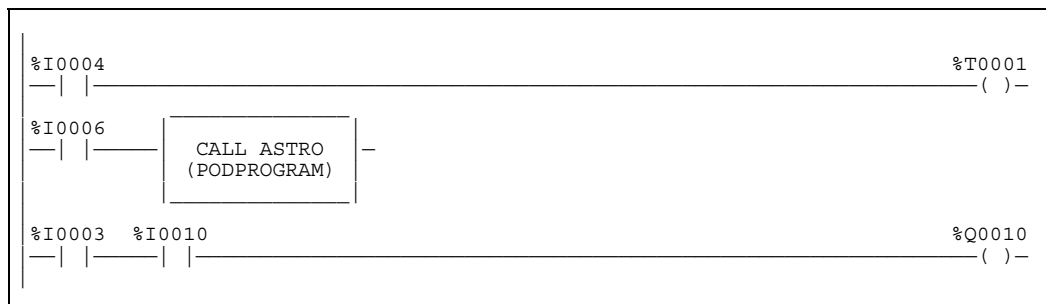
Dodatkowo, podprogramy mogą być też wywoływane z poziomu innych podprogramów. Podprogramy mogą być nawet wywoływane rekurencyjnie.



Sterownik dopuszcza jedynie osiem zagnieżdżonych (jedno wewnątrz drugiego) wywołań, a po przekroczeniu tego limitu następuje zarejestrowanie błędu działania sterownika i zatrzymanie go (przejście do trybu **STOP/FAULT**). Przy obliczaniu poziomu wywołania podprogramów, program główny traktowany jest jako poziom 1.

Wywoływanie podprogramów

Podprogram jest wykonywany po wywołaniu z bloku głównego programu lub z poziomu innego podprogramu.



Po ustawieniu kursora na instrukcji CALL i naciśnięciu klawisza F10 (Zoom), na ekranie wyświetlona zostanie drabina logiczna podprogramu.

Procedury wykonywane okresowo

Podprogramy wykonywane okresowo można wykorzystywać w systemach z jednostką centralną CPU 340 lub nowszą, wersja 4.20 lub nowsza. Należy pamiętać o następujących ograniczeniach:

1. Bloki funkcyjne przełączników czasowych (TMR, ONDTR i OFDTR) nie są realizowane poprawnie w podprogramach wykonywanych okresowo. Zastosowanie bloków funkcyjnych DOIO w podprogramach wykonywanych okresowo z odwołaniami do wyspecjalizowanych modułów wejścia/ wyjścia (HSC, PowerMate, APM, Genius, itp.) powoduje przerwanie komunikacji pomiędzy tymi modułami a jednostką centralną. Zmienne systemowe %S1 (FST_SCN - pierwszy cykl pracy sterownika) i %S2 (LST_SCN - ostatni cykl pracy sterownika) podczas wykonywania takiego podprogramu nie mają określonego stanu. Podprogramy wykonywane okresowo nie mogą być wywoływane z poziomu innych podprogramów, jak również nie można z nich wywoływać innych podprogramów.
2. Opóźnienie wykonania podprogramu (czyli maksymalna zwłoka czasowa pomiędzy planowanym momentem wykonania podprogramu a rzeczywistym wykonaniem podprogramu) wynosi około 35 ms, jeśli w kasecie głównej nie ma zainstalowanego modułu programowalnego koprocatora (PCM), modułu komunikacyjnego (CMM) lub modułu koprocatora wizualizacji (ADC). Jeśli którykolwiek z tych modułów znajduje się w kasecie głównej (nawet jeśli nie jest używany i nie jest skonfigurowany), opóźnienie może wynieść nawet 2.25 ms. Z tego powodu nie jest zalecane stosowanie podprogramów wykonywanych okresowo w systemach z wymienionymi modułami.

Typy zmiennych

Dane wykorzystywane w programie sterującym przechowywane są jako zmienne rejestrowe lub zmienne dyskretne.

Tabela 2-4. Zmienne rejestrowe

Typ	Opis
%R	Zmienna 16-bitowa oznaczająca rejestr, w którym można przechowywać dane programu sterującego (np. wyniki obliczeń). Symbolowi powinien towarzyszyć adres rejestru (np.%R00201).
%AI	Przedrostek %AI oznacza rejestr wejścia analogowego. Po przedrostku podawany jest adres rejestru (np.%AI0015). Zmienna ta przechowuje wartość jednego rejestru wejścia analogowego lub inną wartość.
%AQ	Przedrostek %AQ oznacza rejestr wyjścia analogowego. Po przedrostku podawany jest adres rejestru (np.%AQ0056). Zmienna ta przechowuje wartość jednego rejestru wyjścia analogowego lub inną wartość.

Uwaga

Wartości wszystkich zmiennych rejestrowych są przechowywane w przypadku wyłączenia zasilania sterownika i po ponownym włączeniu pozostają takie same.

Tabela 2-5. Zmienne dyskretne

Typ	Opis
%I	Zmienne poprzedzone tym przedrostkiem reprezentują fizyczne wejścia dyskretne. Po symbolu podawany jest adres zmiennej w tabeli stanu wejść (np.%I00121). Przechowywany jest w niej aktualny stan wszystkich wejść sterownika, odczytany z modułów wejściowych podczas ostatniego odczytu. Adres zmiennej jest przyporządkowywany modułom wejść dyskretnych za pomocą oprogramowania konfiguracyjnego lub programatora ręcznego. Do momentu przyporządkowania adresu zmiennej moduł wejść nie przesyła żadnych danych do sterownika. Zmienne typu %I mogą posiadać pamięć stanu (tzn. zachowywać poprzednią wartość po wyłączeniu zasilania sterownika i ponownym jego włączeniu)*.
%Q	Zmienne poprzedzone tym przedrostkiem reprezentują fizyczne wyjścia dyskretne. Oprogramowanie Logicmaster 90-30/20/Micro posiada funkcję do sprawdzania wielokrotnego wykorzystywania zmiennych %Q w przełącznikach i parametrach wyjściowych. Począwszy od wersji 3 oprogramowania, można wybrać żądany poziom kontroli (SINGLE, WARN MULTIPLE lub MULTIPLE). Więcej informacji o tym mechanizmie podano w podręczniku <i>Logicmaster 90-30/20/Micro Programming Software User's Manual, GFK-0466K</i> . Po symbolu podawany jest adres zmiennej w tabeli stanu wyjść (np.%Q00016). Przechowywany jest w niej aktualny stan wszystkich wyjść sterownika, ostatnio ustawionych przez program sterujący. Wartości te są przesyłane do modułów wyjść podczas fazy obsługi wyjść. Adres zmiennej jest przyporządkowywany modułom wyjść dyskretnych za pomocą oprogramowania konfiguracyjnego lub programatora ręcznego. Do momentu przyporządkowania adresu zmiennej, nie są przesyłane żadne dane do modułu. Zmienne typu %Q mogą posiadać pamięć stanu (tzn. zachowywać poprzednią wartość po wyłączeniu zasilania sterownika i ponownym jego włączeniu). *
%M	Zmienne poprzedzone tym przedrostkiem reprezentują wewnętrzne zmienne dyskretne programu sterującego. Oprogramowanie Logicmaster 90-30/20/Micro posiada funkcję do sprawdzania wielokrotnego wykorzystywania zmiennych %Q w przełącznikach i jako parametry wyjściowe. Począwszy od wersji 3 oprogramowania, można wybrać żądany poziom kontroli (SINGLE, WARN MULTIPLE lub MULTIPLE). Szczegółowe informacje na ten temat podano w podręczniku GFK-0466. Zmienne typu %M mogą posiadać pamięć stanu (tzn. zachowywać poprzednią wartość po wyłączeniu zasilania sterownika i ponownym jego włączeniu). *
%T	Zmienne poprzedzone tym przedrostkiem reprezentują zmienne chwilowe. Zmienne te nie są sprawdzane pod kątem wielokrotnego wykorzystywania w przełącznikach. Zmienne typu %T mogą być stosowane w celu uniknięcia konfliktu przełączników przy posługiwaniu się funkcjami wytnij/ wklej i funkcjami zapisu/ dołączenia do pliku. Ponieważ pamięć przeznaczona na te zmienne nie jest podtrzymywana po wyłączeniu zasilania lub po przejściu z trybu RUN do STOP lub STOP do RUN , zmienne te nie mogą być wykorzystywane w przełącznikach z pamięcią.

* Posiadanie przez zmienną pamięci stanu uzależnione jest od rodzaju przełącznika. Więcej informacji podano w punkcie "Pamięć stanu", strona 2-21.

Tabela 2-5. Zmienne dyskretne - kontynuacja

Typ	Opis
%S	<p>Przedrostek %S umieszczany jest przed zmiennymi systemowymi. Umożliwiają one dostęp do danych systemowych, takich jak informacje o błędach działania sterownika czy pracy modułów wejść/wyjść. Wśród zmiennych systemowych można wyróżnić cztery grupy, oznaczone odpowiednio symbolami %S, %SA, %SB oraz %SC.</p> <p>Zmienne %S, %SA, %SB i %SC mogą być przypisane do dowolnych styków.</p> <p>Zmienne %SA, %SB i %SC mogą być przypisane do przekaźników z pamięcią stanu -(M)-.</p> <p>Zmienne %S mogą być wykorzystane jako parametry wejściowe (typu słowo lub słowo bitowe) bloku funkcyjnego.</p> <p>Zmienne %SA, %SB i %SC mogą być wykorzystane jako parametry wejściowe lub wyjściowe (typu słowo lub słowo bitowe) bloku funkcyjnego.</p>
%G	<p>Przedrostek %G umieszczany jest przed zmiennymi globalnymi. Umożliwiają one dostęp do danych wspólnie wykorzystywanych przez kilka sterowników. Zmienne typu %G zawsze posiadają pamięć stanu i mogą być przypisywane do styków i przekaźników z pamięcią stanu. Nie mogą być natomiast przypisywane do przekaźników bez pamięci.</p>

Bity zmiany stanu i wymuszania wartości

Zmienne typu %I, %Q, %M i %G posiadają bity zmiany stanu (transition) i wymuszenia wartości (override). Zmienne typu %T, %S, %SA, %SB i %SC posiadają bity zmiany stanu, lecz nie posiadają bitów wymuszenia wartości. Bity zmiany stanu wykorzystywane są przez liczniki i przekaźniki uaktywniane zboczem sygnału. Należy zwrócić uwagę, że liczniki nie korzystają z tego samego typu bitów zmiany stanu co przekaźniki. Bity zmiany stanu są zapamiętywane wraz ze zmienną.

Jednostki centralne CPU 331 i wyższe dopuszczają możliwość wymuszenia wartości bitów z zewnątrz. Po ustawieniu wartości takiego bitu, wartość odpowiadającej mu zmiennej nie może zostać zmieniona przez program sterujący ani też przez urządzenie wejściowe; a jedynie poprzez rozkaz programatora.

W jednostkach centralnych 323, 313 i Micro nie jest dostępna możliwość trwałego ustawienia wartości bitów sterujących z zewnątrz.

Pamięć stanu

Zmienna posiada pamięć stanu, jeśli jej wartość jest zapamiętywana po wyłączeniu sterownika. Sterowniki serii 90 przechowują program sterujący, tabele błędów działania i informacje diagnostyczne, zablokowane wartości zmiennych, wartości zmiennych rejestrowych (typu %R, %AI, %AQ), wartości zmiennych dyskretnych (typu %I, %S, %G, bity - znaczniki błędów działania) oraz wartości zmiennych dyskretnych typu %M i %Q w przypadku, gdy są one przypisane do przekaźników z pamięcią. Wartości zmiennych %T nie są zapamiętywane. Chociaż, zgodnie z podanymi powyżej informacjami, pamiętana jest wartość zmiennych %SC, domyślnie, nie jest pamiętana wartość dla zmiennych %S, %SA i %SB.

Zmienne typu %Q i %M, przypisane do przekaźników bez pamięci, nie posiadają pamięci stanu (tzn. ich wartość nie jest pamiętana po przejściu sterownika z trybu **STOP** do trybu **RUN**). Przekaźniki bez pamięci to przekaźniki o następujących symbolach: -(I)-, -(J)-, -(S)- oraz -(R)-.

Jeżeli zmienne typu %Q i %M zostaną przypisane do przekaźników z pamięcią lub wykorzystane jako zmienne wyjściowe bloków funkcyjnych, ich wartość jest przechowywana po wyłączeniu zasilania sterownika oraz po przejściu z trybu pracy **RUN** w tryb **STOP** i ponownie w tryb **RUN**. Przekaźniki z pamięcią to przekaźniki o symbolach: -(M)-, -(N)-, -(SM)- oraz -(RM)-.

Stan zapamiętanej i przechowywanej wartości zmiennej typu %Q lub %M zostaje określony w miejscu jej ostatniego wykorzystania. Przykładowo, jeżeli zmienna %Q0001 zostanie przypisana do przekaźnika z pamięcią, jej wartość będzie pamiętana. Nie mniej jednak, jeżeli zmienna %Q0001 zostanie przypisana do przekaźnika bez pamięci, nie będzie ona posiadała pamięci stanu.

Typy danych

Tabela 2-6. Typy danych

Typ	Nazwa	Opis	Format zapisu
INT	Liczba całkowita ze znakiem	Liczby całkowite ze znakiem zajmują 16 bitów pamięci i są zapisywane w formacie dopełnienia do dwóch dla liczb ujemnych. Zakres: od -32768 do +32768.	<p>Rejestr 1</p>
DINT	Liczba całkowita podwójnej precyzji ze znakiem	Liczby całkowite podwójnej precyzji ze znakiem są przechowywane w 32 bitach pamięci i są zapisywane w formacie dopełnienia do dwóch dla liczb ujemnych. (Bit 32 to bit znaku.) Zakres: od -2 147 483 647 do +2,147,483,867.	<p>Rejestr 2 Rejestr 1</p>
BIT	Bit	Dana zajmująca najmniejszą komórkę pamięci. Może przyjmować wartość 1 lub 0. Słowo bitowe może mieć długość N.	
BYTE	Bajt	Dana zawierająca 8 bitów. Zakres: od 0 do 255 (0 do ff w systemie heksadecymalnym).	
WORD	Word	Słowo zajmuje 16 kolejnych bitów pamięci sterownika, ale w przeciwieństwie do ciągu bitów reprezentującego w pamięci liczbę, bity mogą być niezależne od siebie odczytywane i zmieniane. Każdy bit posiada swój własny stan logiczny (1 lub 0). Zakres wartości: 0 do FFFF (w systemie heksadecymalnym)	<p>Rejestr 1</p>
DWORD	Słowo podwójnej długości	Słowo podwójnej długości posiada taką samą charakterystykę jak typ WORD, jedna różnica to wykorzystywanie do przechowywania 32 kolejnych bitów, a nie 16 kolejnych bitów.	<p>Rejestr 2 Rejestr 1</p>
BCD-4	Czterocyfrowa liczba dziesiętna zakodowana w formacie BCD	Czterocyfrowe liczby dziesiętne zakodowane w formacie BCD zajmują 16 bitów pamięci. Każda z czterech cyfr tej liczby jest zakodowana w czterech bitach i może reprezentować cyfrę z zakresu od 0 do 9. Zakres wartości: od 0 do 9999.	<p>Rejestr 1</p>
REAL	Liczba rzeczywista	Liczby rzeczywiste zajmują 32 kolejne bity pamięci (w rzeczywistości są to dwie kolejne komórki pamięci po 16 bitów każda). Zakres wartości: ± 1.401298E-45 do ± 3.402823E+38.	<p>Rejestr 2 Rejestr 1</p>

S = Bit znaku (0 = dodatni, 1 = ujemny).

Zmienne systemowe

Zmienne typu %S, %SA, %SB oraz %SC są zmiennymi systemowymi. Każda z tych zmiennych posiada nazwę pomocniczą. Przykładowe nazwy zmiennych systemowych sterujących podstawą czasu generatora sygnału prostokątnego to: T_10MS, T_100MS, T_SEC oraz T_MIN. Przykładowe nazwy innych zmiennych, często wykorzystywanych przy programowaniu, to: FST_SCN, ALW_ON i ALW_OFF.

Uwaga

Wartości zmiennych systemowych %S można jedynie odczytywać, nie można ich zmieniać. Można jednakże zmieniać wartości zmiennych %SA, %SB i %SC.

Poniżej podano zmienne systemowe, które można wykorzystywać w programie sterującym. Podczas wprowadzania elementów logicznych można używać zarówno adresów zmiennych, jak i nazw pomocniczych. Więcej informacji na temat błędów i sposobów ich usuwania podano w Rozdziale 3, "Błędy".

nazwy specjalne nie mogą być stosowane w innym kontekście.

Tabela 2-7. Zmienne systemowe

Zmienna	Nazwa	Definicja
%S0001	FST_SCN	Jeżeli bieżący cykl jest pierwszym cyklem pracy sterownika, ustawiana jest wartość 1.
%S0002	LST_SCN	Jeżeli bieżący cykl jest ostatnim cyklem pracy sterownika, ustawiana jest wartość 0.
%S0003	T_10MS	Podstawa czasu generatora sygnału prostokątnego: 0,01 s.
%S0004	T_100MS	Podstawa czasu generatora sygnału prostokątnego: 0,1 s.
%S0005	T_SEC	Podstawa czasu generatora sygnału prostokątnego: 1.0 s.
%S0006	T_MIN	Podstawa czasu generatora sygnału prostokątnego: 1.0 minuta.
%S0007	ALW_ON	Zmienna, której wartość jest zawsze równa 1.
%S0008	ALW_OFF	Zmienna, której wartość jest zawsze równa 0.
%S0009	SY_FULL	Jeżeli tabela błędów działania sterownika jest wypełniona całkowicie, ustawiana jest wartość 1. Wartość zmiennej jest ustawiana ponownie na 0, gdy z tabeli zostanie usunięta część informacji lub cała zawartość tabeli zostanie wykasowana.
%S0010	IO_FULL	Jeżeli tabela błędów działania układów wejść/wyjść jest wypełniona całkowicie, ustawiana jest wartość 1. Wartość zmiennej jest ustawiana ponownie na 0, gdy z tabeli zostanie usunięta część informacji lub cała zawartość tabeli zostanie wykasowana.
%S0011	OVR_PRE	Wymuszona z zewnątrz zmiana wartości zmiennej typu %I, %Q, %M lub %G.
%S0013	PRG_CHK	Ustawiana na 1 w czasie obliczania sumy kontrolnej programu sterującego.
%S0014	PLC_BAT	Jeżeli bateria w jednostkach centralnych z oprogramowaniem systemowym ver. 4 lub nowsza jest rozładowana, zmienna przyjmuje wartość 1. Zmienna ta uaktualniana jest jeden raz w ciągu cyklu.

Tabela 2-7. Zmienne systemowe - kontynuacja

Zmienna	Nazwa	Definicja
%S0017	SNP_XACT	Urządzenie Master protokołu SNP-X rozpoczęło komunikację z jednostką centralną.
%S0018	SNP_X_RD	Urządzenie Master protokołu SNP-X odczytało dane z jednostki centralnej.
%S0019	SNP_X_WT	Urządzenie Master protokołu SNP-X zapisało dane do jednostki centralnej.
%S0020		Zmienna ustawiana na 1 po pomyślnym wykonaniu relacji matematycznej operującej na danych typu rzeczywistego. Jeżeli parametr wejściowy ma wartość NaN (Not a Number), bit ten jest zerowany.
%S0032		Zarezerwowana dla oprogramowania.
%SA0001	PB_SUM	Suma kontrolna wyliczona dla programu sterującego nie jest równa wartości porównawczej. Jeśli błąd ten został spowodowany przez chwilową usterkę, można go usunąć poprzez ponowne zapisanie programu w pamięci sterownika. Jeśli natomiast uszkodzona została pamięć RAM, należy wymienić jednostkę centralną sterownika.
%SA0002	OV_SWP	Poprzedni cykl pracy sterownika trwał dłużej niż zostało to wyszczególnione przez użytkownika. Wartość tej zmiennej jest ustawiana ponownie na 0, gdy czas trwania cyklu pracy sterownika nie przekroczy zadanej wartości. Wartość 0 jest również ustawiana w czasie przejścia z trybu zatrzymania STOP do trybu pracy RUN . Zmienna ta zachowuje ważność wyłącznie w trybie o stałym czasie trwania.
%SA0003	APL_FLT	Błąd w działaniu programu sterującego. Wartość tej zmiennej zostaje ponownie ustawiona na 0, gdy sterownik przechodzi z trybu STOP tryb RUN .
%SA0009	CFG_MM	Wartość tej zmiennej ustawiana jest na 1, jeżeli konfiguracja zapisana w oprogramowaniu, sprawdzana podczas zapisu konfiguracji lub włączania zasilania układu, nie jest zgodna z konfiguracją sprzętową. Wartość tej zmiennej jest ustawiana ponownie na 0, gdy konfiguracje zgadzają się.
%SA0010	HRD_CPU	Wartość tej zmiennej ustawiana jest na 1, po wykryciu przez procedury diagnostyczne uszkodzenia jednostki centralnej. Wartość tej zmiennej jest ustawiana ponownie na 0 po wymianie jednostki centralnej.
%SA0011	LOW_BAT	Niskie napięcie baterii zasilających. Wartość tej zmiennej jest ustawiana ponownie na 0 po wymianie baterii.
%SA0014	LOS_IOM	Przerwanie komunikacji pomiędzy jednostką centralną a modułem wejść/wyjść. Wartość tej zmiennej ustawiana jest ponownie na 0 po wymianie modułu, wyłączeniu i ponownym włączeniu zasilania.
%SA0015	LOS_SIO	Przerwanie komunikacji z jednym ze specjalizowanych modułów. Wartość tej zmiennej ustawiana jest ponownie na 0 po wymianie modułu, wyłączeniu i ponownym włączeniu zasilania.
%SA0019	ADD_IOM	Wartość tej zmiennej jest ustawiana na 1, po dodaniu do kasety modułu wejść/wyjść. Wartość 0 jest ponownie ustawiana po wyłączeniu i ponownym włączeniu zasilania sterownika, gdy zapamiętana konfiguracja odpowiada rzeczywistej.
%SA0020	ADD_SIO	Wartość tej zmiennej jest ustawiana na 1, jeżeli do kasety dodano jeden z opcjonalnych modułów. Wartość 0 jest ponownie ustawiana po wyłączeniu i ponownym włączeniu zasilania sterownika, gdy zapamiętana konfiguracja odpowiada rzeczywistej.
%SA0027	HRD_SIO	Wartość tej zmiennej ustawiana jest na 1, jeżeli wykryte zostało uszkodzenie sprzętowe w jednym z modułów opcjonalnych. Wartość tej zmiennej ustawiana jest ponownie na 0 po wymianie modułu, wyłączeniu i ponownym włączeniu zasilania.
%SA0031	SFT_SIO	Wartość tej zmiennej ustawiana jest na 1, jeżeli wykryty został niepoprawialny błąd programowy w jednym z modułów opcjonalnych. Wartość 0 jest ponownie ustawiana po wyłączeniu i ponownym włączeniu zasilania sterownika, gdy zapamiętana konfiguracja odpowiada rzeczywistej.
%SB0010	BAD_RAM	Ustawiana na 1, jeżeli jednostka centralna wykryła uszkodzenie pamięci RAM podczas rozruchu. Ustawiana na 0, jeżeli jednostka centralna nie stwierdzi uszkodzenia pamięci RAM podczas rozruchu.

Tabela 2-7. Zmienne systemowe - kontynuacja

Zmienna	Nazwa	Definicja
%SB0011	BAD_PWD	Ustawiana na 1, w przypadku wprowadzenia błędnego hasła dostępu do sterownika. Ustawiana na 0 po usunięciu zawartości tabeli błędów działania sterownika.
%SB0013	SFT_CPU	Ustawiana na 1, jeżeli jednostka centralna sterownika wykryje nienaprawialny błąd w oprogramowaniu. Ustawiana na 0 po usunięciu zawartości tabeli błędów działania sterownika.
%SB0014	STOR_ER	Ustawiana na 1, jeżeli w czasie operacji ładowania programu sterującego z programatora wykryty zostanie błąd. Ustawiana na 0 po udanym załadowaniu programu.
%SC0009	ANY_FLT	Ustawiana na 1 w przypadku wykrycia jakichkolwiek błędów działania sterownika lub układów wejść/wyjść. Ustawiana na 0, gdy obie tabele błędów działania nie zawierają komunikatów o błędach.
%SC0010	SY_FLT	Ustawiana na 1 jeżeli wystąpiło uszkodzenie powodujące wprowadzenie nowej pozycji do tabeli błędów działania sterownika. Ustawiana na 0 jeżeli tabela ta nie zawiera żadnych komunikatów o błędach.
%SC0011	IO_FLT	Ustawiana na 1 jeżeli wystąpił jakikolwiek błąd działania układów wejść/wyjść sterownika, zapisywany w tabeli błędów działania wejść/wyjść. Ustawiana na 0 jeżeli tabela ta nie zawiera żadnych komunikatów o błędach.
%SC0012	SY_PRES	Wartość tej zmiennej jest równa 1 przez cały czas, jeśli tylko w tabeli błędów działania sterownika znajduje się choć jedna informacja o błędzie. Ustawiana na 0 jeżeli tabela ta nie zawiera żadnych komunikatów o błędach.
%SC0013	IO_PRES	Wartość tej zmiennej jest równa 1 przez cały czas, jeśli tylko w tabeli błędów działania układów wejść/wyjść znajduje się choć jedna informacja o błędzie. Ustawiana na 0 jeżeli tabela ta nie zawiera żadnych komunikatów o błędach.
%SC0014	HRD_FLT	Wartość ustawiana na 1 w momencie wystąpienia uszkodzenia sprzętowego. Ustawiana na 0, gdy obie tabele błędów działania nie zawierają komunikatów o błędach.
%SC0015	SFT_FLT	Wartość ustawiana na 1 w momencie pojawienia się błędu oprogramowania. Ustawiana na 0, gdy obie tabele błędów działania nie zawierają komunikatów o błędach.

Uwaga: Wszystkie pozostałe zmienne %S, nie wymienione w powyższej tabeli są zmiennymi zarezerwowanymi i nie powinny być wykorzystywane w programie sterującym.

Podstawowe informacje o instrukcjach języka drabinkowego

Każdy szczebel drabiny logicznej programu sterującego składa się z jednego lub większej liczby elementów logicznych. Mogą to być zarówno proste przekaźniki, jak i bardziej złożone bloki funkcyjne.

Format przekaźników i styków

Oprogramowanie narzędziowe udostępnia kilka typów przekaźników oraz styków. Elementy te zapewniają podstawową kontrolę przepływu sygnału w programie sterującym. Przykładem mogą być styki otwarte i przekaźniki o stykach zamkniętych. Każdy z tych styków i przekaźników posiada jedno wejście i jedno wyjście. Zastosowane łączenie, umożliwiają sterowanie przepływem sygnału.

Do każdego przekaźnika i styku musi być przypisana zmienna, która jest wprowadzana równocześnie z wprowadzeniem tego elementu do szczebla. W przypadku styków, zmienna ta reprezentuje bit w pamięci sterownika, którego wartość określa przepływ sygnału przez styk. W poniższym przykładzie, jeśli zmienna %I0122 ma wartość 1, przez styk przepływa sygnał:

```
%I0122
- | -
```

W przypadku przekaźników, zmienna ta reprezentuje bit w pamięci sterownika, którego wartość jest ustawiana w zależności od tego, czy do przekaźnika doprowadzany jest sygnał. W poniższym przykładzie, jeśli do przekaźnika doprowadzany jest sygnał, wartość zmiennej %Q0004 jest ustawiana na 1:

```
%Q0004
- ( ) -
```

Oprogramowanie oraz programator ręczny posiadają możliwość kontroli wielokrotnego wykorzystania tych samych zmiennych typu %Q i %M, przypisanych do różnych przekaźników i wyjść bloków funkcyjnych.

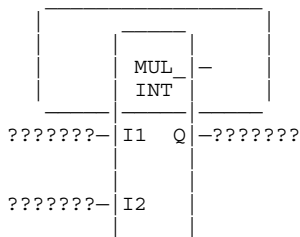
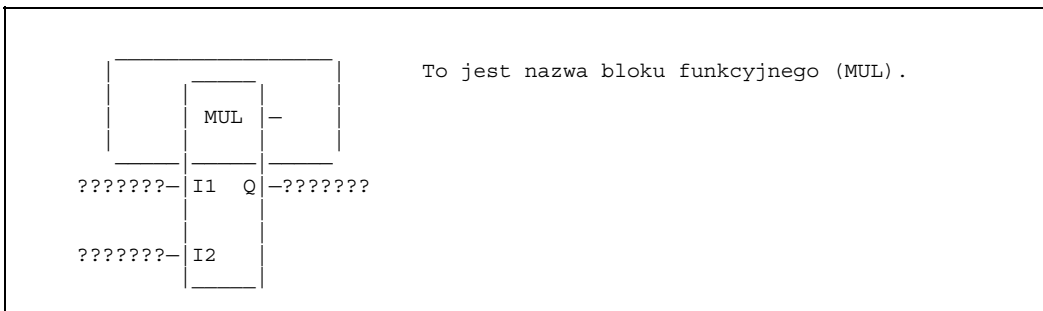
Format bloków funkcyjnych

Niektóre funkcje stosowane w programie sterującym mają bardzo prostą postać, np. funkcja MCR, pokazana poniżej:

```
- [ MCR ] -
```

Większość funkcji posiada bardziej złożoną postać. Wymagają one podania kilku parametrów, wykorzystywanych przez funkcję podczas jej wykonywania.

Format takiego elementu pokazano na przykładzie funkcji mnożenia dwóch liczb (MUL). Jego parametry są typowe dla wielu funkcji. Górna część bloku zawiera zwykle nazwę funkcji. W niektórych przypadkach może tu być również podany typ danych, w niniejszym przypadku liczba całkowita ze znakiem.

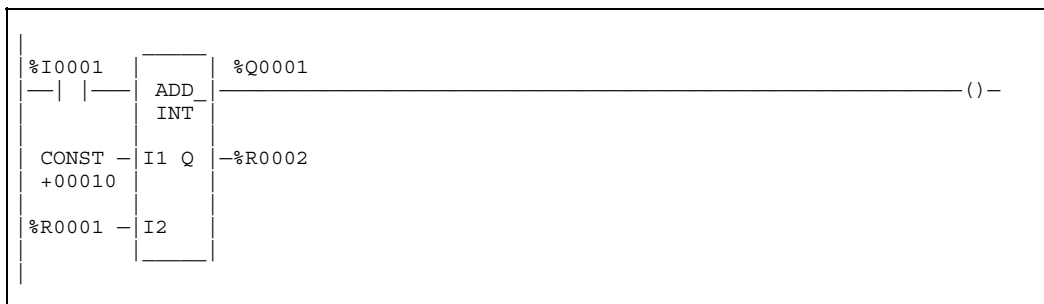


Wiele funkcji umożliwia wybór lub zmianę typu danych, na których funkcja wykonuje operacje, już po wyborze funkcji. Przykładowo, typ danych INT dla funkcji MUL można zmienić na DINT. Typy danych opisano we wcześniejszej części niniejszego rozdziału.

Parametry bloków funkcyjnych

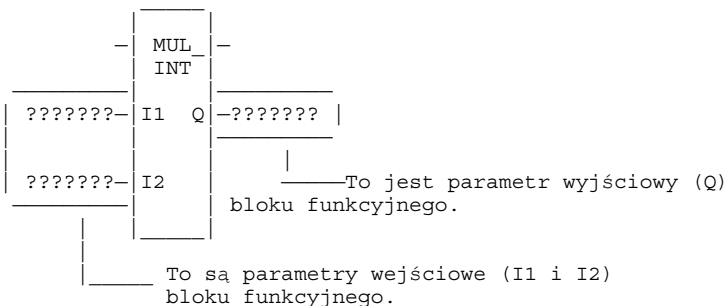
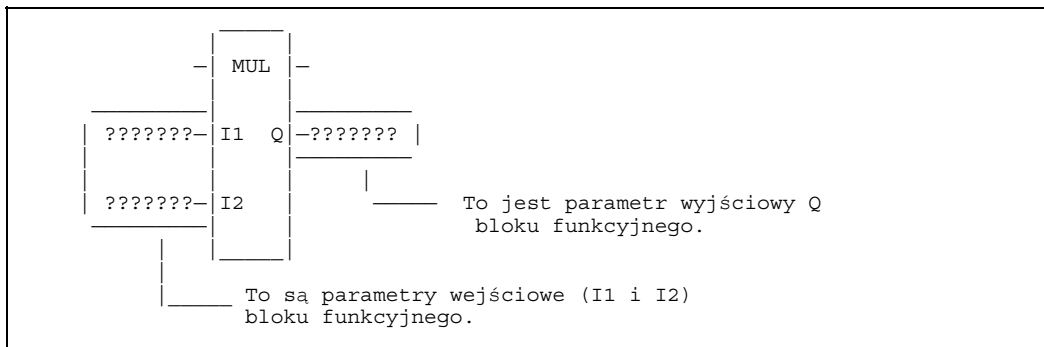
Każda linia dochodząca do lewej strony bloku funkcyjnego reprezentuje parametr wejściowy. Parametrami wejściowymi mogą być stałe lub zmienne. Stała jest wartością zadaną w postaci jawnej. Zmienna jest identyfikowana poprzez jej adres.

W poniższym przykładzie (blok funkcyjny ADD) parametr wejściowy I1 jest stałą, natomiast parametr I2 jest zmienną.



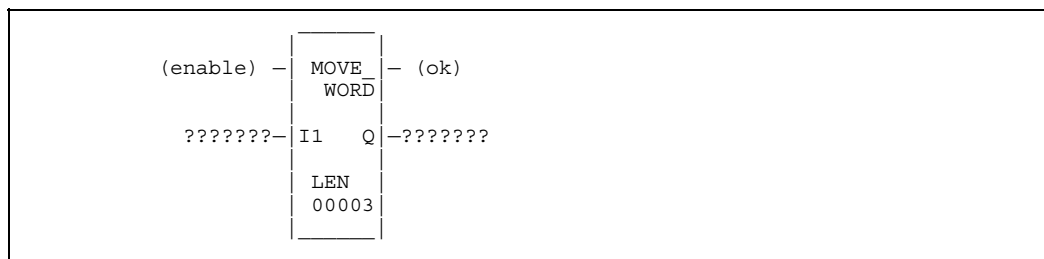
Każda linia wychodząca z prawej strony bloku funkcyjnego reprezentuje parametr wyjściowy. Parametrami wyjściowymi bloków funkcyjnych mogą być wyłącznie zmienne, nie mogą to być stałe.

Jeżeli z lewej strony bloku funkcyjnego umieszczony jest znak zapytania, można wprowadzić bezpośrednio samą wartość, adres pod którym znajduje się wartość lub zmienną, określającą adres, gdzie znajduje się wartość parametru. Jeżeli znak zapytania umieszczony jest z prawej strony bloku funkcyjnego, należy podać adres, pod który ma być zapisana wartość wyjściowa z bloku funkcyjnego albo zmienną, określającą adres, pod który ma być zapisana wartość wyjściowa z bloku funkcyjnego.

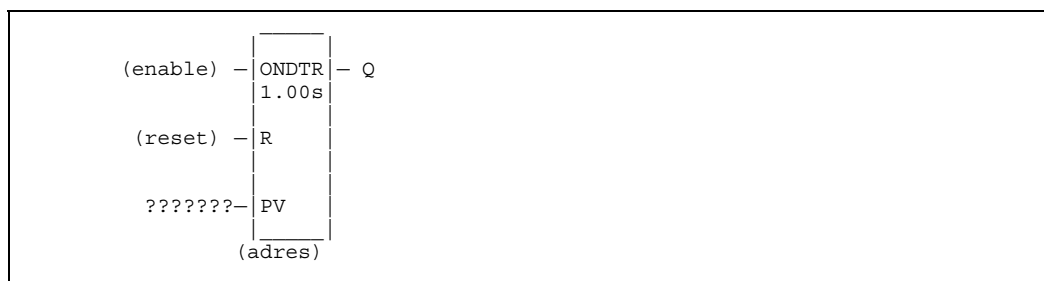


Większość bloków funkcyjnych nie powoduje zmian wartości wejść, natomiast zapisuje wynik wykonanej operacji do parametru wyjściowego.

W przypadku funkcji operujących na tabelach, można wybrać ilość elementów, na których będzie wykonywana operacja. W zamieszczonym poniżej bloku funkcyjnym, parametr LEN podaje liczbę słów do przesunięcia.

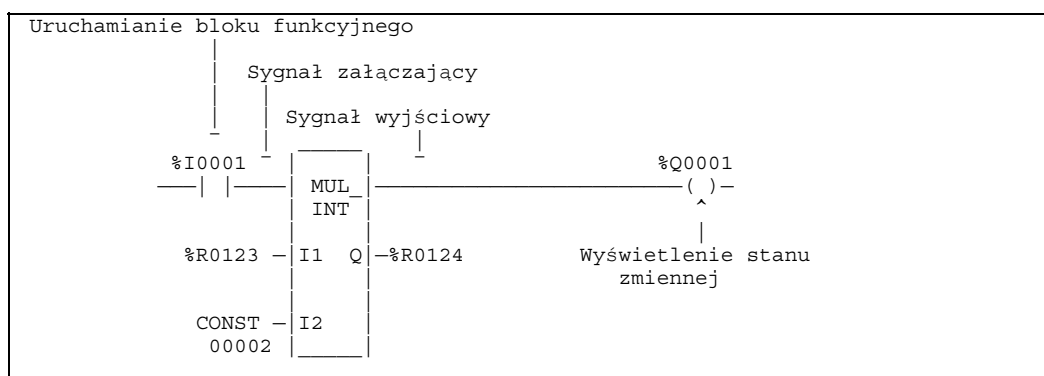


Bloki funkcyjne liczników, przekaźników czasowych, BITSEQ i PID wymagają wprowadzenia adresu początkowego obszaru pamięci o długości 3 słów, w którym przechowywane są: wartość zadana, wartość bieżąca oraz słowo sterujące.



Przepływ sygnału przez blok funkcyjny

Sygnal doprowadzany jest do bloku funkcyjnego z jego lewej strony, u samej góry. Dopływ sygnału do bloku funkcyjnego jest zawsze sterowany za pomocą styku; jeśli blok funkcyjny jest połączony z szyną sygnałową za pośrednictwem styku zwartej na stałe, to blok ten jest wykonywany w każdym cyklu pracy sterownika.



Uwaga

Bloki funkcyjne nie mogą być bezpośrednio podłączone do lewej strony ekranu. Jeżeli funkcja ma być wywoływana w każdym cyklu, można skorzystać ze zmiennej %S7, ALW_ON i styku otwartego, połączonych z symboliczną linią zasilającą po lewej stronie ekranu.

Sygnal wyjściowy bloku funkcyjnego umieszczony jest z prawej strony tego bloku, u samej góry. Może on być wykorzystywany przez inne elementy logiczne programu lub przekaźniki (opcjonalnie). Bloki funkcyjne przesyłają sygnał wyjściowy po pomyślnym wykonaniu operacji.

Część 3: Rozruch i wyłączanie sterownika

Rozruch sterowników serii 90-30 może przebiegać na dwa sposoby: jest to tzw. rozruch "na zimno" lub "na gorąco". Jednostka centralna zazwyczaj przeprowadza rozruch "na zimno", jednakże w przypadku jednostek CPU 331 i wyższych, jeśli czas pomiędzy wyłączeniem zasilania a ponownym jego włączeniem jest krótszy niż 5 sekund, przeprowadzany jest rozruch "na gorąco".

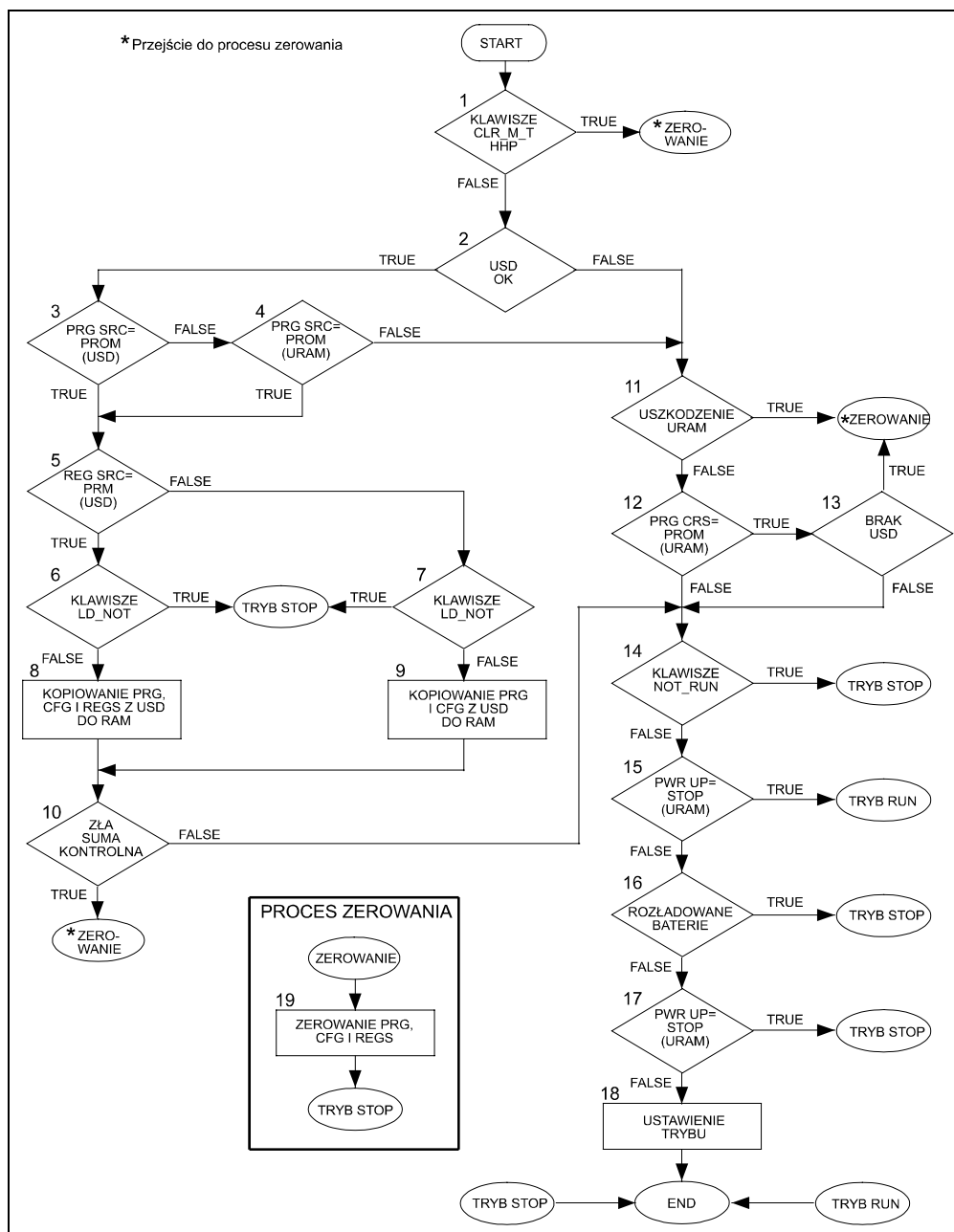
Rozruch sterownika

Rozruch "na zimno" składa się z wyszczególnionego poniżej ciągu operacji. Rozruch "na gorąco" różni się od rozruchu "na zimno" jedynie pominięciem operacji opisanych w kroku 1.

1. Jednostka centralna przeprowadza autodiagnostykę. W kroku tym zawarta jest między innymi kontrola części pamięci RAM podtrzymywanej bateryjnie w celu stwierdzenia, czy zawiera ona poprawne dane.
2. Jeśli sterownik wyposażony jest w pamięć EPROM lub EEPROM i konfiguracja nakazuje wykorzystanie zawartości tej pamięci, to jest ona kopiowana do pamięci RAM. Jeśli pamięć EPROM lub EEPROM nie jest zainstalowana, zawartość pamięci RAM pozostaje bez zmian.
3. Jednostka centralna sprawdza każde gniazdo kasyety w celu określenia, jakie moduły są zainstalowane.
4. Konfiguracja sprzętowa jest porównywana z konfiguracją zaprogramowaną (powinny być one takie same). Wszystkie wykryte niezgodności traktowane są jako błędy. Również brak modułów lub obecność modułów nie zadeklarowanych traktowane są jako błędy działania i zostają zakomunikowane użytkownikowi.
5. Jeśli nie zaprogramowano konfiguracji, jednostka centralna wybiera automatycznie konfigurację standardową.
6. Jednostka centralna wybiera kanał komunikacyjny pomiędzy nią a wyspecjalizowanymi modułami dodatkowymi.
7. Ostatecznie, na podstawie konfiguracji jednostki centralnej, określany jest tryb pracy sterownika (**RUN** lub **STOP**) dla pierwszego cyklu. W trybie **RUN** cykl realizowany jest zgodnie z opisem podanym w punkcie poświęconym przejściu z trybu zatrzymania **STOP** do trybu pracy **RUN**. Na Rys. 2-5 zamieszczonym na następnej stronie przedstawiono sekwencje decyzji podejmowanych przez jednostkę centralną w celu określenia, czy ma być kopiowana zawartość pamięci PROM oraz jaki tryb pracy sterownika ma być wybrany (**STOP** lub **RUN**).

Uwaga

Kroki 2 do 7 opisane powyżej nie mają zastosowania w odniesieniu do sterownika serii 90 Micro. Więcej informacji o sekwencjach rozruchu i wyłączania dla sterownika Micro podano w Rozdziale 5 podręcznika *Series 90 Micro PLC User's Manual* (GFK-1065).



Rysunek 2-5. Sekwencja rozruchu

Przed wykonaniem instrukcji START zamieszczonej na schemacie blokowym rozruchu, realizowane są czynności diagnostyczne, w czasie których testowane są różne urządzenia zewnętrzne, wykorzystywane przez jednostkę centralną oraz testowana jest pamięć RAM. Po zakończeniu czynności diagnostycznych następuje inicjalizacja wewnętrznych struktur danych i urządzeń zewnętrznych wykorzystywanych przez jednostkę centralną. Następnie jednostka centralna sprawdza, czy nie została uszkodzona pamięć RAM wykorzystywana przez użytkownika. Jeżeli stwierdzone zostało uszkodzenie pamięci RAM, następuje usunięcie programu sterującego, wyzerowanie wszystkich rejestrów użytkownika oraz wprowadzenie konfiguracji domyślnej.

OBJAŚNIENIE skrótów stosowanych na schemacie

PRG = Program sterujący

CFG = Konfiguracja

REGS = Rejestry użytkownika (zmiennie %I, %Q, %M, %G, %R, %AI i %AQ).

USD = Urządzenie do zapisywania danych (pamięć) użytkownika, może to być zarówno EPROM jak i urządzenie z możliwością ponownego zapisu (EEPROM).

URAM = Pamięć podtrzymywana, zawierająca program sterujący, konfigurację i rejestry.

PEŁNA WERSJA SKRÓCONYCH NAPISÓW:

- (1) (1) KLAWISZE CLR_M_T - Czy w czasie rozruchu wciśnięto klawisze <CLR> i <M_T> programatora ręcznego w celu wyczyszczenia pamięci URAM?
- (2) (2) USD OK ? - Czy zainstalowana jest pamięć USD (User Storage Device), może jej nie być wyłącznie w modelach korzystających z pamięci EEPROM oraz czy zawarte w niej informacje są poprawne?
- (3) (3) PRG SRC= PROM (USD) - Czy parametr PRG SRC z zestawu USD ma wartość Prom, co oznacza wczytywanie programu sterującego i konfiguracji z urządzenia USD?
- (4) (3) PRG SRC= PROM (USD) - Czy parametr PRG SRC z zestawu USD ma wartość Prom, co oznacza wczytywanie programu sterującego i konfiguracji z urządzenia USD?
- (5) (5) REG SRC= PRM (USD) - Czy parametr REG SRC z zestawu USD ma wartość Prom, co oznacza wczytywanie programu sterującego i konfiguracji z urządzenia USD?
- (6 & 7) (6 & 7) KLAWISZE LD_NOT - Czy w czasie rozruchu wciśnięto klawisze <LD> i <NOT> programatora ręcznego w celu uniemożliwienia wczytywania programu sterującego i rejestrów z urządzenia USD?
- (8) (8) KOPIOWANIE PRG, CFG I REGS Z USD DO RAM - Kopiowanie programu sterującego, konfiguracji i rejestrów z urządzenia USD do URAM.
- (9) (9) KOPIOWANIE PRG I CFG Z USD DO RAM - Kopiowanie programu sterującego i konfiguracji z urządzenia USD do URAM.
- (10) (10) ZŁA SUMA KONTROLNA - Czy sumy kontrolne dla programu sterującego i konfiguracji, wczytane z urządzenia USD są poprawne?
- (11) (11) USZKODZENIE URAM - Czy zawartość pamięci URAM jest poprawna? Przyczyna uszkodzenia zawartości może być włączenie bez włożonej baterii lub z baterią wyczerpaną. Zakłócenia tego typu mogą być również spowodowane przez uaktualnienie oprogramowania systemowego.
- (12) (3) PRG SRC= PROM (USD) - Czy parametr PRG SRC z zestawu USD ma wartość Prom, co oznacza wczytywanie programu sterującego i konfiguracji z urządzenia USD?
- (13) BRAK USD - Czy zainstalowana jest pamięć USD? Dotyczy wyłącznie modelu korzystających z pamięci EEPROM.
- (14) KLAWISZE NOT_RUN - Czy w czasie rozruchu wciśnięto na programatorze ręcznym klawisze i w celu bezwarunkowego rozpoczęcia pracy w trybie zatrzymania (**STOP**).
- (15) PWR UP=STOP (URAM) - Czy parametr PWR UP z zestawu URAM ma wartość run ?
- (16) ROZŁADOWANE BATERIE - Czy baterie są rozładowane?
- (17) PWR UP=STOP (URAM) - Czy parametr PWR UP z zestawu URAM ma wartość **STOP**?
- (18) (18) USTAWIENIE TRYBU - Ustawienie trybu RUN/STOP na tryb obowiązujący w momencie wyłączania sterownika.
- (19) ZEROWANIE PRG, CFG I REGS - Zerowanie programu sterującego, konfiguracji i rejestrów.

Uwaga

Pierwsza część schematu 2-5, zamieszczonego na poprzedniej stronie, nie dotyczy sterowników serii 90 Micro. Więcej informacji o sekwencjach rozruchu i wyłączania dla sterownika Micro podano w Rozdziale 5 podręcznika *Series 90 Micro PLC User's Manual* (GFK-1065).

Wyłączanie sterownika

Wyłączenie systemowe (samoczynne) sterownika następuje w przypadku zaniku napięcia w sieci zasilającej prądu przemiennego na okres dłuższy niż jeden jego cykl lub gdy napięcie zasilania sterownika prądem stałym (nominalnie wynoszące 5 V) spadnie poniżej 4.9 V.

Część 4: Zegary

Sterowniki serii 90-30 posiadają kilka rodzajów zegarów: zegar odmierzający czas pracy sterownika, zegar czasu rzeczywistego (modele 331, 340/341, 351/352 i 28-o punktowy sterownik Micro), zegar wyłączający oraz zegar cyklu pracy o stałym czasie trwania. Trzy bloki funkcyjne przekaźników czasowych to przekaźnik czasowy załączający, przekaźnik czasowy załączający z pamięcią oraz przekaźnik czasowy wyłączający. Dostępne są również cztery generatory sygnału prostokątnego o podstawie 0.01 s, 0.1 s, 1.0 s i 1 minuty.

Zegar odmierzający czas pracy sterownika

Zegar ten wykorzystuje 100 ms odcinki czasu do odmierzania czasu, który upłynął od momentu rozruchu jednostki centralnej. Zegar ten nie przechowuje odmierzonej wartości po wyłączeniu zasilania, jest on zerowany przy każdym rozruchu. Co sekundę zegar komunikuje się z jednostką centralną w celu zapisania odmierzonej wartości (w sekundach). Odmierzanie sekund może trwać do około 100 lat od momentu włączenia tego zegara.

Ponieważ zegar odmierzający czas pracy sterownika jest wykorzystywany przez oprogramowanie systemowe oraz przez bloki funkcyjne przekaźników czasowych, nie można go wyzerować z poziomu programu sterującego ani z programatora. Jednakże program sterujący może odczytać aktualną wartość czasu odmierzonego przez zegar za pomocą bloku funkcyjnego SVCREQ #16.

Zegar czasu rzeczywistego

Ten zegar sprzętowy posiadają tylko jednostki centralne sterowników serii 90-30, modele CPU331 i wyższe oraz 28-o punktowe sterowniki Micro. Udostępnia on następujące dane:

- Rok (dwie cyfry)
- Miesiąc
- Dzień miesiąca
- Godzina
- Minuta
- Sekunda
- Dzień tygodnia

Zegar czasu rzeczywistego jest zasilany z baterii i pracuje w przypadku awarii lub wyłączenia zasilania sterownika. Aby jednak skorzystać z odmierzanych przezeń wartości, należy dokonać jego inicjalizacji. Program może odczytywać i ustawiać ten zegar za pomocą funkcji SVCREQ#7. Zegar można również odczytywać i ustawiać za pomocą oprogramowania do konfigurowania jednostki centralnej. Nie można zmienić wskazań zegara czasu rzeczywistego za pomocą programatora, jeżeli włączona jest funkcja zabezpieczenia pamięci przed dokonaniem zmian za pomocą wyłącznika kluczewego.

Zegar podtrzymujący aktualną datę i czas jest zaprojektowany zgodnie z kalendarzem. Uwzględniane są automatycznie lata przestępne, do roku 2079.

Zegar wyłączający

Zegar wyłączający sterowników serii 90-30 przeznaczony jest do wykrywania błędów krytycznych, które powodują znaczną wydłużenie czasu trwania cyklu pracy sterownika. Wartość progowa, powodujące dla zegara wyłączającego wynosi 200 ms (500 milisekund dla jednostek centralnych serii 35x i 36x) i nie może zostać zmieniona. Zegar ten zostaje wyzerowany na początku każdego cyklu pracy.

W sterownikach 90-30, w jednostkach centralnych serii 331 i niższych, po przekroczeniu wartości progowej gaszona jest dioda LED "OK", jednostka centralna zostaje zatrzymana, wyjścia przechodzą w stan standardowy, przyjmowany samoczynnie. Wszystkie mikroprocesory modułów zostają zatrzymane i ustaje wszelka komunikacja. Aby uruchomić sterownik ponownie, należy wyłączyć i włączyć zasilanie kasety sterownika zawierającej jednostkę centralną. W jednostkach centralnych sterowników serii 90-20, 90-30

i 90 Micro, modele 340 i wyższe, zegar wyłączający powoduje wyzerowanie jednostki centralnej, wykonanie programu logicznego dla rozruchu, wygenerowanie błędu o uszkodzeniu zegara wyłączającego i przejście do trybu **STOP**.

Zegar czasu trwania przerwy w zasilaniu sterownika

Zegar czasu trwania przerwy w zasilaniu sterownika podaje czas trwania braku zasilania sterownika. Po wyłączeniu zasilania, zegar ten jest zerowany, po czym rozpoczyna zliczanie. Po włączeniu zasilania, zliczanie czasu jest wstrzymywane, a naliczona wartość zostaje zapamiętana. Wskazania tego zegara można odczytać za pomocą bloku funkcyjnego SVCREQ #29, opisanego w rozdziale 12.

Uwaga

Funkcja ta dostępna jest wyłącznie w jednostkach centralnych 331 i wyższych sterowników serii 90-30.

Zegar cyklu pracy o stałym czasie trwania

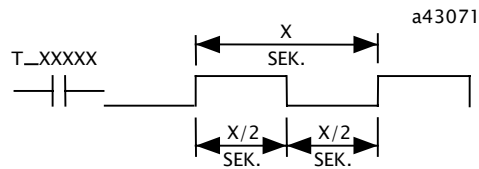
Zegar ten kontroluje czas trwania cyklu, jeżeli sterownik serii 90-30 pracuje w trybie o stałym czasie trwania (**CONSTANT SWEEP TIME**). W trybie tym, czas realizacji każdego cyklu jest taki sam. Zwykle, w przypadku większości programów sterujących, fazy odczytywania wejść, wykonywania części logicznej programu sterującego oraz ustawiania wyjść nie wymagają takiej samej ilości czasu w każdym z cykli. Stały czas trwania cyklu jest ustawiany za pomocą programatora i może wynosić od 5 do wartości progowej zegara wyłączającego (domyślnie 100 milisekund).

Jeśli cykl nie zakończy się przed upływem zadanego czasu trwania, a czas trwania poprzedniego cyklu nie został przekroczony, w tabeli błędów działania sterownika zostaje umieszczony komunikat alarmowy. Na początku kolejnego cyklu wartość zmiennej systemowej OV_SWP zostaje ustawiona na 1. Wartość tej zmiennej zostaje ponownie ustawiona na 0, gdy sterownik zostanie przełączony z trybu pracy o stałym czasie trwania (**CONSTANT SWEEP TIME**) na inny lub gdy czas trwania ostatniego cyklu jest krótszy od zdefiniowanego.

Generator sygnału prostokątnego

Sterowniki serii 90 posiadają cztery styki generatora sygnału prostokątnego, o podstawach czasu 0.01 s, 0.1 s, 1 s oraz 1 min. Stan tych styków nie zmienia się w czasie wykonywania cyklu. Poprzez styki te wysyłany jest sygnał o takim samym czasie włączenia jak i wyłączenia. Do styków tych przypisane są zmienne systemowe (odpowiednio T_10MS, T_100MS, T_SEC oraz T_MIN).

Działanie generatora objaśnia zamieszczony poniżej rysunek.



Rysunek 2-6. Schemat pracy generatora sygnału prostokątnego

Część 5: Mechanizmy zabezpieczające przed dostępem

Mechanizmy zabezpieczające przed dostępem osób nieupoważnionych, dostępne w sterownikach serii 90-30, 90-20 i Micro, mają na celu uniemożliwienie dokonania zmian w sterowniku osobom nieupoważnionym. W sterowniku dostępne są cztery poziomy uprawnień. Na pierwszym poziomie, który jest zawsze dostępny, można odczytywać dane ze sterownika, nie można wprowadzać żadnych zmian w aplikacji. Pozostałe trzy poziomy zabezpieczane są hasłem.

Wraz ze wzrostem poziomu wzrasta zakres zmian, które mogą być wprowadzane. Uprawnienia dostępne na danym poziomie są sumą uprawnień dla tego poziomu, plus uprawnień dostępnych na poziomach niższych. Dostępne są następujące poziomy uprawnień:

Poziom uprawnień	Opis
Poziom 1	Możliwość odczytywania wszystkich danych, za wyjątkiem haseł. Można więc odczytywać pamięć (%I, %Q, %AQ, %R, itp.), tabele błędów oraz wszystkie typy bloków programu (dane, wartości i stałe). Nie jest dostępna możliwość wprowadzania zmian w sterowniku.
Poziom 2	Poziom ten pozwala na zapisywanie danych do pamięci (%I, %R, itp.).
Poziom 3	Poziom ten pozwala na zapisywanie zmian w programie sterującym wyłącznie w trybie zatrzymania STOP .
Poziom 4	Jest to poziom przyjmowany domyślnie dla systemów w których nie ustawiono hasła. Domyślnym poziomem dla systemu z ustawionymi hasłami jest najwyższy poziom bez ustawionego hasła. Poziom ten, najwyższy, pozwala na odczytywanie i zapisywanie danych do wszystkich pamięci, włączając w to hasła, zarówno w trybie pracy RUN jak i w trybie zatrzymania STOP . (Nie można zmieniać konfiguracji w trybie RUN).

Hasła

Każdy poziom uprawnień sterownika posiada swoje hasło. (Nie można ustawić hasła dla 1 poziomu uprawnień). Dla każdego poziomu może być wprowadzone inne hasło, można też wprowadzić takie samo hasło dla więcej niż jednego poziomu. Hasła mogą mieć długość od jednego do czterech znaków ASCII, można je wprowadzać i zmieniać za pomocą oprogramowania lub za pomocą programatora ręcznego.

Poziom uprawnień obowiązuje tak długo, jak długo trwa komunikacja pomiędzy sterownikiem a programatorem. Nie jest konieczne wykonywanie jakichkolwiek działań, nie mniej jednak komunikacja nie może zostać przerwana. W przypadku braku komunikacji przez 15 minut, poziom uprawnień zostaje zmieniony na najwyższy, nie zabezpieczony poziom.

Po podłączeniu do sterownika, oprogramowanie żąda podania statusu zabezpieczeń każdego z poziomów uprawnień. Następnie, oprogramowanie wysyła do sterownika żądanie przejścia do najwyższego, niezabezpieczonego poziomu, nie określając tym samym, który to jest poziom. Jeżeli do sterownika podłączony jest programator ręczny, sterownik przełącza się do najwyższego, niezabezpieczonego poziomu dostępu.

Zmiana poziomu uprawnień

W celu zmiany poziomu dostępu należy wprowadzić za pomocą programatora nowy poziom dostępu oraz hasło dla tego poziomu. Żądanie przejścia na inny poziom dostępu jest odrzucane, jeżeli wprowadzone hasło nie jest zgodne z hasłem zapisanym dla tego poziomu w tabeli haseł dostępu. W sytuacji takiej poziom dostępu nie jest zmieniany, nie są wprowadzane żadne zmiany. Jeżeli podjęta zostanie próba wprowadzenia zmian w sterowniku za pomocą programatora ręcznego, bez posiadania uprawnień, programator wyświetli komunikat o błędzie, informujący o braku dostępu.

Blokowanie/odblokowywanie podprogramów

Za pomocą funkcji udostępnianych przez oprogramowanie, można blokować i odblokowywać podprogramy. Dostępne są dwa rodzaje blokad:

Typ blokady	Opis
Blokada podglądu	Po zablokowaniu, nie można wyświetlić zawartości podprogramu.
Blokada edycji	Po zablokowaniu podprogram może być wyświetlony, ale nie można go edytować.

Zablokowanie podglądu i edycji podprogramów można usunąć z poziomu edytora deklaracji bloków programu, o ile nie jest to trwała blokada podglądu i edycji.

W podprogramach z zablokowanym podglądem można korzystać z funkcji edycyjnych do szukania i zastępowania. Jeżeli szukany tekst zostanie odnaleziony w podprogramie z zablokowanym podglądem, zamiast programu sterującego wyświetlony zostanie następujący komunikat:

```
Found in locked block <block_name> (Continue/Quit)
(Znaleziono miejsce wystąpienia w zablokowanym bloku <nazwa_bloku>
(kontynuacja/koniec))
```

lub

```
Cannot write to locked block <block_name> (Continue/Quit)
(Nie można zapisywać do zablokowanego bloku <nazwa_bloku> (kontynuacja/koniec))
```

Operacja szukania można kontynuować (Continue) lub zakończyć (Quit).

Można usuwać zawartość i same foldery z zablokowanymi podprogramami. Jeżeli folder zawiera zablokowane podprogramy, bloki te pozostają zablokowane w czasie wywoływania funkcji oprogramowania Copy, Backup i Restore.

Trwała blokada podprogramów

Poza blokadą podglądu (view lock) i edycji (edit lock) dostępne są dwa rodzaje trwałych blokad. Po założeniu trwałej blokady podglądu (permanent view lock), wszelkie próby podglądu podprogramu są odrzucane. Po założeniu trwałej blokady edycji (permanent edit lock), wszelkie próby edycji podprogramu są odrzucane.

Ostrzeżenie

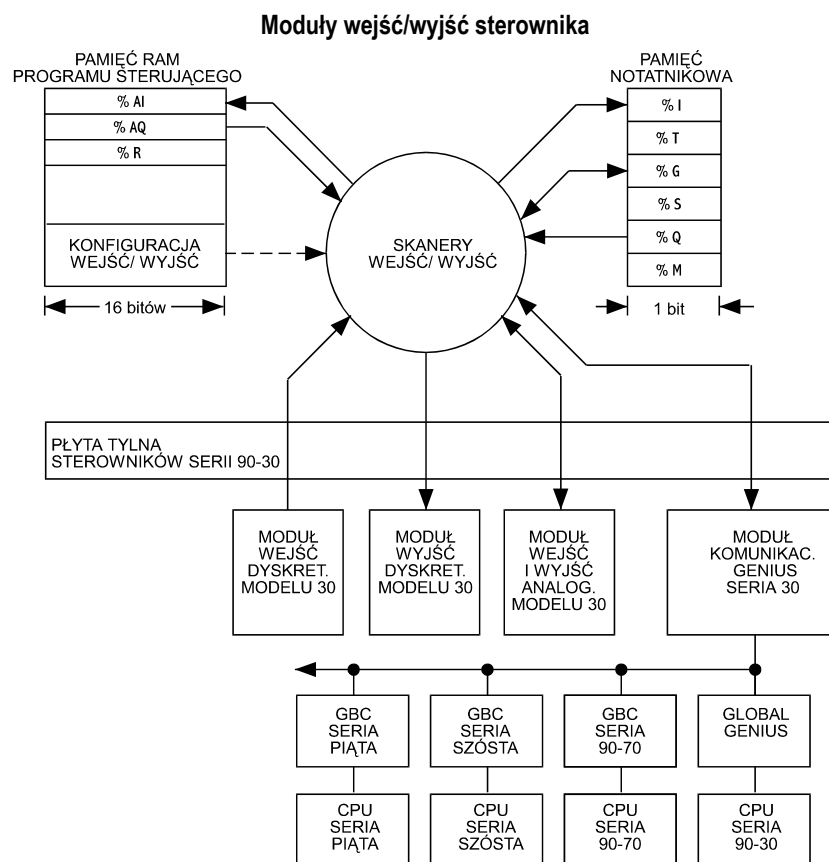
Po założeniu blokad trwałych, w odróżnieniu od normalnych blokad podglądu i edycji, nie można ich już usunąć.

Po założeniu trwałej blokady edycji (PERMANENT EDIT LOCK), można ją jedynie zmienić na trwałą blokadę podglądu (PERMANENT VIEW LOCK). Jeżeli została założona blokada PERMANENT VIEW LOCK, nie można jej zmienić na żadną, innego typu blokadę.

Część 6: Moduły wejść/wyjść sterowników serii 90-30, 90-20 i Micro

Moduły wejść i wyjść sterownika zapewniają łączność z urządzeniami i osprzętem użytkownika. Moduły wejść/wyjść dla sterowników serii 90-30 nazywane są skrótowo modułami wejść/wyjść serii 90-30. Moduły wejść/wyjść serii 90-30 są instalowane bezpośrednio w gniazdach kasety głównej lub gniazdach kaset rozszerzających w sterownikach serii 90-30, modele 331 i wyższe. Jednostki centralne 331, 340 i 341 obsługują do 49 modułów wejść/wyjść serii 90-30 (5 kaset). Jednostki centralne 351 i 352 obsługują do 79 modułów wejść/wyjść serii 90-30 (8 kaset). Jednostki centralne serii 90-30, model 311 i 313 z kasetą wyposażoną w 5 gniazd obsługują do 5 modułów wejść/wyjść serii 90-30; a model 323 z kasetą wyposażoną w 10 gniazd obsługuje do 10 modułów wejść/wyjść serii 90-30.

Na zamieszczonym poniżej rysunku pokazano strukturę wejść i wyjść dla sterowników serii 90-30.



Rysunek 2-7. Moduły wejść/wyjść sterowników serii 90-30

Uwaga

Na zamieszczonym powyżej rysunku pokazano typową dla sterowników 90-30 strukturę systemu wejść i wyjść. Moduły wyspecjalizowane i dodatkowe nie są obsługiwane w czasie skanowania wejść/wyjść; są one obsługiwane w fazie komunikacji systemowej. Dodatkowe informacje na temat modułów wejść/wyjść sterowników serii 90-20 podano w podręczniku *Series 90™-20 Programmable Controller User's Manual* (GFK-0551). Więcej informacji na temat struktury wejść/wyjść sterowników Micro podano w podręczniku *Series 90 Micro PLC User's Manual* (GFK-1065).

Moduły wejść/wyjść sterowników serii 90-30

Dostępnych jest następujących pięć rodzajów modułów wejść i wyjść serii 90-30: moduły wejść dyskretnych, wyjść dyskretnych, wejść analogowych, wyjść analogowych oraz specjalizowane moduły dodatkowe. W zamieszczonej poniżej tabeli zestawiono moduły wejść/wyjść serii 90-30, numery katalogowe, liczbę punktów wejść/wyjść oraz krótki opis każdego z modułów.

Uwaga

Aktualna oferta może nie zawierać wszystkich wymienionych poniżej modułów wejść/wyjść. W celu uzyskania informacji o aktualnej ofercie prosimy o skontaktowanie się z lokalnym dystrybutorem lub autoryzowanym sprzedawcą sterowników GE Fanuc. Parametry techniczne oraz sposób podłączania każdego z modułów wejść/wyjść sterowników 90-30 zamieszczono w podręczniku Series 90-30 I/O Module Specifications Manual, GFK-0898.

Tabela 2-8. Moduły wejść/wyjść serii 90-30

Numer katalogowy	Liczba punktów	Opis	Nr podręcznika
<i>Moduły wejść dyskretnych</i>			
IC693MDL230	8	120 VAC, izolowane	GFK-0898
IC693MDL231	8	240 VAC, izolowane	GFK-0898
IC693MDL240	16	120 VAC	GFK-0898
IC693MDL241	16	24 VAC/ DC, logika ujemna	GFK-0898
IC693MDL630	8	24 VDC, logika dodatnia	GFK-0898
IC693MDL632	8	125 VDC, logika dodatnia/ ujemna	GFK-0898
IC693MDL633	8	24 VDC, logika ujemna	GFK-0898
IC693MDL634	8	24 VDC, logika dodatnia/ ujemna	GFK-0898
IC693MDL640	16	24 VDC, logika dodatnia	GFK-0898
IC693MDL641	16	24 VDC, logika ujemna	GFK-0898
IC693MDL643	16	24 VDC, logika dodatnia, FAST	GFK-0898
IC693MDL644	16	24 VDC, logika ujemna, FAST	GFK-0898
IC693MDL645	16	24 VDC, logika dodatnia/ ujemna	GFK-0898
IC693MDL646	16	24 VDC, logika dodatnia/ ujemna, FAST	GFK-0898
IC693MDL652	32	24 VDC, logika dodatnia/ ujemna	GFK-0898
IC693MDL653	32	24 VDC, logika dodatnia/ ujemna, FAST	GFK-0898
IC693MDL654	32	5/12 VDC (TTL), logika dodatnia/ ujemna	GFK-0898
IC693MDL655	32	24 VDC, logika dodatnia/ ujemna	GFK-0898
IC693ACC300	8/16	Symulator wejść	GFK-0898

Tabela 2-8. Moduły wejść/wyjść serii 90-30 - kontynuacja

Numer katalogowy	Liczba punktów	Opis	Nr podręcznika
<u>Moduły wyjść dyskretnych</u>			
IC693MDL310	12	120 V AC, 0.5 A	GFK-0898
IC693MDL330	8	120/240 V AC, 2A	GFK-0898
IC693MDL340	16	120 V AC, 0.5 A	GFK-0898
IC693MDL390	5	120/240 V AC, Izolowane, 2A	GFK-0898
IC693MDL730	8	12/24 V DC, logika dodatnia, 2A	GFK-0898
IC693MDL731	8	12/24 V DC, logika ujemna, 2A	GFK-0898
IC693MDL732	8	12/24 V DC, logika dodatnia, 0.5 A	GFK-0898
IC693MDL733	8	12/24 V DC, logika ujemna, 0,5A	GFK-0898
IC693MDL734	6	125 V DC, logika ujemna, 2 A	GFK-0898
IC693MDL740	16	12/24 V DC, logika dodatnia, 0.5 A	GFK-0898
IC693MDL741	16	12/24 V DC, logika ujemna, 0,5A	GFK-0898
IC693MDL742	16	12/24 V DC, logika dodatnia, 1 A	GFK-0898
IC693MDL750	32	12/24 V DC, logika ujemna	GFK-0898
IC693MDL751	32	12/24 V DC, logika dodatnia, 0,3A	GFK-0898
IC693MDL752	32	5/24 V DC (TTL), logika ujemna, 0.5 A	GFK-0898
IC693MDL753	32	12/24 V DC, logika dodatnia/ ujemna, 0.5 A	GFK-0898
IC693MDL930	8	Wyjścia przekaźnikowe zwierne, 4 A, izolowane	GFK-0898
IC693MDL931	8	Wyjścia przekaźnikowe rozwierne, izolowane	GFK-0898
IC693MDL940	16	Wyjścia przekaźnikowe zwierne, 2 A, izolowane	GFK-0898
<u>Moduły wejść/wyjść</u>			
IC693MDR390	8/8	Wejście 24 V DC, wyjścia przekaźnikowe	GFK-0898
IC693MAR590	8/8	Wejście 120 V AC, wyjścia przekaźnikowe	GFK-0898
<u>Moduły analogowe</u>			
IC693ALG220	4 kanał.	Wejście analogowe, napięciowy	GFK-0898
IC693ALG220	4 kanał.	Wejście analogowe, prądowy	GFK-0898
IC693ALG222	16	Wejście analogowe, napięciowy	GFK-0898
IC693ALG223	16	Wejście analogowe, prądowy	GFK-0898
IC693ALG390	2 kanał.	Wyjście analogowe, napięciowy	GFK-0898
IC693ALG391	2 kanał.	Wyjście analogowe, prądowy	GFK-0898
IC693ALG392	8 kanał.	Wyjście analogowe, prądowo-napięciowy	GFK-0898
IC693ALG442	4/2	Analogowy, kombinowany, prądowo- napięciowy	GFK-0898

Tabela 2-8. Moduły wejść/wyjść serii 90-30 - kontynuacja

Numer katalogowy	Opis	Nr podręcznika
<i>Moduły dodatkowe</i>		
IC693APU300	Licznik impulsów o wysokiej częstotliwości	GFK-0293
IC693APU301	Moduł pozycjonowania, 1 osiowy pracujący w trybie standardowym lub nadążnym	GFK-0781 GFK-0840
IC693APU302	Moduł Power Mate APM 2 osiowy, nadążny	GFK-0781
IC693MCS001/2	Moduł Power Mate APM 2 osiowy, standardowy System sterowania ruchem Power Mate J (1 i 2 osiowy)	GFK-0840 GFK-1256
IC693APU305	Moduł procesora wejść/wyjść dla sygnałów szybkodziennych	GFK-1028
IC693CMM321	Moduł sieciowy Ethernet	GFK-1084
IC693ADC311	Koprocesor wyświetlacza alfanumerycznego	GFK-0521
IC693BEM331	Moduł komunikacyjny Genius Bus Controller	GFK-1034
IC693BEM320	Interfejs dla sterowników 90-70 i układów CNC (slave)	GFK-0631
IC693BEM321	Interfejs dla sterowników 90-70 i układów CNC (master)	GFK-0823
IC693CMM311	Moduł komunikacyjny CMM	GFK-0582
IC693CMM301	Moduł komunikacyjny Genius	GFK-0412
IC693CMM302	Moduł komunikacyjny Enhanced Genius	GFK-0695
IC693PCM300	Moduł programowalnego koprocesora PCM, 160 KB (35 Kb dla programu w języku MegaBasic)	GFK-0255
IC693PCM301	Moduł programowalnego koprocesora PCM, 192 KB (47 Kb dla programu w języku MegaBasic)	GFK-0255
IC693PCM311	Moduł programowalnego koprocesora PCM, 640 KB (190 Kb dla programu w języku MegaBasic)	GFK-0255

Formaty danych wejściowych/wyjściowych

Stan wejść i wyjść dyskretnych jest przechowywany w postaci bitów w pamięci cache sterownika. Dane dotyczące wejść i wyjść analogowych są zapamiętywane jako słowa bitowe w części pamięci RAM przeznaczonej do tego celu.

Domyślne wartości wyjść modułów serii 90-30

Przy rozruchu sterownika, moduły wyjść dyskretnych sterowników 90-30 ustawiają samoczynnie wartości wyjść na 0. Pozostają one w tym stanie do momentu obsługi wyjść w pierwszym cyklu pracy sterownika. Wyjścia analogowe można skonfigurować za pomocą przełącznika umieszczonego w module tak, aby ustawione zostały one przy rozruchu na 0 lub zatrzymywały ostatnią wartość. Moduły wyjść analogowych mogą być zasilane z zewnętrznego źródła zasilania, co pozwala na pamiętanie wartości przy wyłączonym zasilaniu sterownika.

Dane diagnostyczne

W komórkach pamięci adresowanych przez określone zmienne systemowe (patrz tabela zmiennych systemowych) znajdują się informacje dotyczące ewentualnych błędów działania modułów wejść/wyjść sterownika lub niezgodności konfiguracji ze stanem faktycznym. Nie są dostępne informacje diagnostyczne odnośnie poszczególnych punktów wejść lub wyjść. Więcej informacji na temat obsługi błędów podano w Rozdziale 3, "Błędy"

Dane globalne

Dane globalne Genius

Sterowniki serii 90-30 pozwalają na wzajemne komunikowanie się jednostek centralnych, poprzez moduł komunikacyjny GENIUS i dyskretne zmienne globalne (%G). Sterownik szyny Genius, IC693BEM331 w jednostkach centralnych CPU, wersja 5 lub późniejsza oraz moduł komunikacyjny Genius, IC693CMM302, umożliwiają przesyłanie do 128 bajtów danych do innych sterowników lub komputerów. Mogą one również odbierać dane o wielkości do 128 bajtów z/do 30 innych sterowników Genius, połączonych w sieć. Dane mogą być przesyłane i odbierane nie tylko jako dyskretne zmienne globalne %G, ale również jako zmienne innego typu. Oryginalny moduł komunikacyjny Genius, IC693CMM301, umożliwia korzystanie wyłącznie ze zmiennych globalnych %G, a pozwala na przesyłanie do 32 bitów poprzez adresy szyny szeregowej od SBA 16 do 23. Nie jest zalecane stosowanie tego modułu ponieważ moduł komunikacyjny Enhanced Genius posiada znacznie zwiększone możliwości i przepustowość.

Dane globalne mogą być wymieniane pomiędzy sterownikami Series Five, Series Six i serii 90, połączonych do tej samej magistrali Genius.

Dane globalne

W jednostkach centralnych CPU364 (oprogramowanie systemowe wer. 9.0 i nowsze) można przysyłać dane siecią Ethernet za pomocą jednego z dwóch, wbudowanych portów Ethernet (ale nie przez obydwa jednocześnie). Jednostki te wyposażone są w port AAUI i port 10BaseT. Spośród jednostek centralnych serii 90-30, tylko jednostka centralna CPU364 (oprogramowanie systemowe wer. 9.10 lub nowsza) obsługuje Dane globalne Ethernet.

Jednostka centralna CPU364 obsługuje Dane globalne Ethernet (EGD – Ethernet Global Data), które są bardzo podobne do Danych globalnych Genius. Podobieństwo wynika z faktu, że jedno urządzenie (producent) może przysyłać dane do jednego lub więcej urządzeń sieciowych (konsumentów). Dane globalne Ethernet nie są obsługiwane przez oprogramowanie Logicmaster 90 (wymagany jest pakiet do programowania sterowników serii 90, pracujący w środowisku Windows).

Moduły wejść/wyjść sterowników serii 90-20

Poniżej przedstawiono moduły wejść/ wyjść oferowane dla sterowników 90-20. Dla każdego modułu podano numer katalogowy, liczbę punktów wejść/ wyjść oraz krótki opis. Moduł wejść/ wyjść jest umieszczony na płycie montażowej wspólnie z zasilaczem. Parametry techniczne i sposób podłączania każdego z modułów omówiono w rozdziale 5 podręcznika *Series 90-20 Programmable Controller User's Manual*, GFK-0551.

Numer katalogowy	Opis	Liczba punktów wejść/wyjść
IC692MAA541	Moduł wejść/wyjść z zasilaczem, wejścia 120 VAC, wyjścia 120 VAC, zasilacz 120 VAC	16 wejść/ 12 wyjść
IC692MDR541	Moduł wejść/wyjść z zasilaczem, wejścia 24 VAC, wyjścia przekaźnikowe, zasilacz 120 VAC	16 wejść/ 12 wyjść
IC692MDR741	Moduł wejść/wyjść z zasilaczem, wejścia 24 VDC, wyjścia przekaźnikowe, zasilacz 240VAC	16 wejść/ 12 wyjść
IC692CPU211	Moduł jednostki centralnej CPU, model CPU 211	Nie dotyczy

Konfigurowanie i programowanie

Konfigurowanie jest procesem polegającym na przypisywaniu adresów logicznych oraz innych parametrów do modułów sprzętowych systemu. Proces ten można zrealizować zarówno przed jak i po programowaniu, za pomocą oprogramowania konfiguracyjnego lub programatora ręcznego, nie mniej jednak zalecane jest przeprowadzenie konfigurowania przed rozpoczęciem programowania. Więcej informacji na temat konfigurowania podano w podręczniku *Programming Software User's Manual*, GFK-0466.

Programowanie polega na utworzeniu programu sterującego dla sterownika. Ponieważ sterowniki serii 90-30, 90-20 i Micro korzystają z tego samego zestawu instrukcji, do ich programowania można wykorzystywać to samo oprogramowanie Logicmaster 90-30. W rozdziałach 4 do 12 opisano instrukcje programowania, które można wykorzystać do budowy programu sterującego dla sterowników serii 90-30 i 90-20.

Instrukcje instalowania oprogramowania Logicmaster 90-30/20/Micro podano w podręczniku *Programming Software User's Manual*, GFK-0466. W podręczniku tym opisano tworzenie, przesyłanie, edycję oraz drukowanie programów.

Rozdział 3

Błędy

Rozdział ten zawiera informacje odnośnie błędów działania sterowników 90-30, 90-20 i Micro. Podano opisy błędów rejestrowanych w tabeli błędów działania sterownika oraz w tabeli błędów układów wejść i wyjść.

Przy omawianiu błędów w niniejszym rozdziale, dla każdego z nich podano opis zamieszczany w tabeli błędów sterownika oraz kategorię błędu w tabeli błędów działania sterownika. W celu uzyskania dodatkowych informacji o błędach należy odszukać w niniejszym podręczniku odpowiedni opis lub kategorię, wyświetlane do ekranie programatora. Dla każdego błędu podawana jest również przyczyna i instrukcje odnośnie usuwania.

Rozdział 3 składa się z następujących punktów:

Punkt	Zawartość	Strona
Obsługa błędów	Typy błędów działania oraz sposób ich wyświetlania w tabelach błędów. Zamieszczono również informacje odnośnie sposobu wyświetlania tabeli błędów działania układów wejść/wyjść i tabeli błędów działania sterownika.	3-2
Tabela błędów działania sterownika	Pełny opis błędów działania sterownika, wraz z instrukcjami ich usuwania.	3-6
Tabela błędów działania układów wejść/wyjść	Opis kategorii błędów Loss of I/O Module i Addition of I/O Module.	3-16

Część 1: Obsługa błędów

Uwaga

Informacje o obsłudze błędów dotyczą systemów programowanych za pomocą oprogramowania Logicmaster 90-30/20/Micro.

Wystąpienie pewnych błędów lub okoliczności, mających wpływ na pracę systemu, sygnalizowane jest w sterownikach serii 90-30, 90-20 i Micro w postaci błędów. Błędy te, jak na przykład brak modułu wejść/wyjść lub kasety, mogą uniemożliwiać sterowanie maszyną lub procesem. Błędy mogą mieć również pozytywny wydźwięk, jak na przykład włączenie do eksploatacji nowego modułu. Można też wyróżnić błędy mające na celu wyłącznie zwrócenie uwagi użytkownika, jak na przykład informacja o niskim stanie naładowania baterii podtrzymującej zawartość pamięci i związanej z tym konieczności jej wymiany.

Alarmy

Wystąpienie pewnych zmian i uszkodzeń sygnalizowane jest w postaci błędów. Jeżeli informacja o uszkodzeniu zostanie odebrana i przetworzona przez jednostkę centralną, uszkodzenie takie nazywane jest alarmem. Oprogramowanie jednostki centralnej zajmujące się obsługą błędów jest nazywane procesorem alarmów. Użytkownik komunikuje się z procesorem alarmów za pomocą oprogramowania. Błędy działania sterownika i układów wejść/wyjść są rejestrowane w odpowiednich tabelach.

Klasy błędów

Sterowniki serii 90-30, 90-20 i Micro są w stanie wykrywać różnego typu błędy. Są to uszkodzenia wewnętrzne, uszkodzenia zewnętrznych modułów wejść/wyjść oraz błędy operacyjne.

Rodzaj błędu	Przykłady
Uszkodzenia wewnętrzne	Nie zgłaszające się moduły Rozładowana bateria podtrzymująca pamięć Błędna suma kontrolna pamięci sterownika.
Uszkodzenia zewnętrznych modułów wejść/wyjść	Brak kasety lub modułu. Dodanie kasety lub modułu.
Błędy operacyjne	Błędy komunikacji. Niezgodność konfiguracji. Próby użycia błędnego hasła dostępu do sterownika.

Uwaga

Więcej informacji na temat obsługi błędów w sterownikach Micro podano w podręczniku *Series 90 Micro PLC User's Manual* (GFK-1065).

Reakcja systemu na błędy

Błędy sprzętowe mogą powodować zatrzymanie sterownika lub mogą być tolerowane (sterownik nadal pracuje). Uszkodzenia zewnętrznych modułów wejść/wyjść mogą być tolerowane przez sterownik, ale może to stanowić zagrożenie dla sterowanego procesu. Błędy operacyjne są normalnie tolerowane. Błędy działania sterowników serii 90-30, 90-20 i Micro posiadają dwa atrybuty:

Atrybut	Opis
Wykorzystywana tabela błędów	Tabela błędów działania układów wejść/wyjść Tabela błędów działania sterownika
Waga błędu	Błąd krytyczny Błąd diagnostyczny Błąd informacyjny

Tabele błędów działania

Błędy związane z układami wejść/wyjść zapisywane są w tabeli błędów działania układów wejść/wyjść, pozostałe błędy zapisywane są w tabeli błędów działania sterownika. W zamieszczonej poniżej tabeli podano grupy błędów, wagi, tabele błędów oraz nazwy dyskretnych zmiennych systemowych %S, których wartość jest zmieniana w przypadku wystąpienia błędów.

Tabela 3-1. Zestawienie błędów

Grupa błędów	Waga błędu	Tabela błędów	Zmienne systemowe			
			io_ft	any_ft	io_pres	los_iom
Loss of or Missing I/O Module	Diagnostyczny	We/Wy	io_ft	any_ft	io_pres	los_iom
Loss of or Missing Option Module	Diagnostyczny	Sterownika	sy_ft	any_ft	sy_pres	los_sio
System Configuration Mismatch	Krytyczny	Sterownika	sy_ft	any_ft	sy_pres	cfg_mm
PLC CPU Hardware Failure	Krytyczny	Sterownika	sy_ft	any_ft	sy_pres	hrd_cpu
Program Checksum Failure	Krytyczny	Sterownika	sy_ft	any_ft	sy_pres	pb_sum
Low Battery	Diagnostyczny	Sterownika	sy_ft	any_ft	sy_pres	low_bat
PLC Fault Table Full	Diagnostyczny	—	sy_full			
I/O Fault Table Full	Diagnostyczny	—	io_full			
Application Fault	Diagnostyczny	Sterownika	sy_ft	any_ft	sy_pres	apl_ft
No User Program	Informacyjny	Sterownika	sy_ft	any_ft	sy_pres	no_prog
Corrupted User RAM	Krytyczny	Sterownika	sy_ft	any_ft	sy_pres	bad_ram
Password Access Failure	Diagnostyczny	Sterownika	sy_ft	any_ft	sy_pres	bad_pwd
PLC Software Failure	Krytyczny	Sterownika	sy_ft	any_ft	sy_pres	sft_cpu
PLC Store Failure	Krytyczny	Sterownika	sy_ft	any_ft	sy_pres	stor_er
Constant Sweep Time Exceeded	Diagnostyczny	Sterownika	sy_ft	any_ft	sy_pres	ov_swp
Unknown PLC Fault	Krytyczny	Sterownika	sy_ft	any_ft	sy_pres	
Unknown I/O Fault	Krytyczny	We/Wy	io_ft	any_ft	io_pres	

Wagi błędów

Błędy można podzielić na krytyczne, diagnostyczne oraz informacyjne.

Wystąpienie błędu krytycznego powoduje zapisanie tego błędu w odpowiedniej tabeli, ustawienie wartości odpowiednich zmiennych systemowych oraz zatrzymanie systemu. W przypadku błędów diagnostycznych, są one zapisywane w odpowiedniej tabeli oraz ustawiania jest wartość zmiennych diagnostycznych. Błędy informacyjne są zapisywane wyłącznie w odpowiedniej tabeli.

Działania podejmowane, w zależności od wagi błędów, przedstawiono w zamieszczonej poniżej tabeli:

Tabela 3-2. Wagi błędów i działania podejmowane w momencie ich wystąpienia

Waga błędu	Podejmowane działania
Błąd krytyczny	Zarejestrowanie błędu w tabeli błędów Ustawienie wartości zmiennych systemowych na 1. Zatrzymanie sterownika (tryb STOP).
Błąd diagnostyczny	Zarejestrowanie błędu w tabeli błędów Ustawienie wartości zmiennych systemowych na 1.
Informacyjny	Zarejestrowanie błędu w tabeli błędów

W przypadku wystąpienia błędu, jednostka centralna podejmuje odpowiednie działania, w zależności od wagi błędu. Nie można zmieniać wagi błędów w sterownikach 90-30, 90-20 i Micro.

Zmienne do obsługi błędów

W sterownikach 90-30 można korzystać ze specjalnych zmiennych systemowych(%S,%Sx), sygnalizujących wystąpienie błędów w pracy systemu. Zmienne te zawierają informacje o typie zarejestrowanego błędu. Zmienna opisująca błąd pozostaje ustawiona (1), aż do momentu wyzerowania sterownika lub zmianie jej wartości przez program sterujący.

Poniżej zamieszczono przykład, pokazujący sposób zerowania zmiennych systemowych informujących o wystąpieniu błędu z poziomu programu sterującego. W przykładzie tym, zmienna light_01 przypisana do przekaźnika jest ustawiana w przypadku przekroczenia dopuszczalnego czasu trwania cyklu. Zwarcie styku %IO359 powoduje wyzerowanie zarówno zmiennej light_01 jak i zmiennej systemowej OV_SWP.

```

| ov_swp light_01
| —] [—————()—
|
| %IO359 ov_swp
| —] [—————(R)—

```

Wartości zmiennych do obsługi błędów

Procesor alarmów pamięta wartość 128 bitów w obszarze pamięci adresowanym przez zmienne systemowe. Zmienne do obsługi błędów można wykorzystywać w celu określenia miejsca wystąpienia błędu oraz jego rodzaju. Zmienne %S, %SA, %SB i %SC posiadają nazwy pomocnicze. Można z nich korzystać w programie sterującym. Pełne zestawienie zmiennych systemowych podano w Rozdziale 2 "Działanie systemu".

Dodatkowe skutki wystąpienia błędów działania

W przypadku wystąpienia jednego z podanych poniżej błędów, podejmowane są działania, opisane w zamieszczonej poniżej tabeli:

Błąd działania	Opis
PLC CPU Software Failure	Po każdorazowym zarejestrowaniu takiego błędu, jednostka centralna natychmiast przechodzi w specjalny tryb pracy ERROR SWEEP . W trybie tym nie są dozwolone jakiegokolwiek operacje. Jedyną metodą anulowania tego trybu jest wyłączenie zasilania sterownika i jego ponowne włączenie.
PLC Sequence Store Failure	Jeśli podczas ładowania programu sterującego do pamięci nastąpi przerwa w komunikacji sterownika z urządzeniem, z którego ładowany jest program lub też nastąpi inne zakłócenie przerywające ładowanie, rejestrowany jest błąd ładowania programu. Tak długo, jak błąd ten obecny jest w systemie, sterownik nie może zostać uruchomiony (nie może przejść w tryb RUN).

Tabela błędów działania sterownika

W tabeli błędów działania sterownika rejestrowane są takie błędy jak nieprawidłowe hasło, niezgodność konfiguracji, błędna suma kontrolna czy błędy komunikacji.

Oprogramowanie może pracować w dowolnym trybie. Jeżeli oprogramowanie znajduje się w trybie **OFFLINE** informacje o błędach nie są wyświetlane. Są one natomiast pokazywane w trybach **ONLINE** i **MONITOR**. W trybie **ONLINE**, można usuwać informacje o błędach (funkcja ta może być zabezpieczona hasłem).

Po usunięciu informacji o błędzie pomimo, że błąd nadal występuje w systemie, nie jest on rejestrowany w tabeli (za wyjątkiem błędu informującego o rozładowaniu baterii).

Tabela błędów działania układów wejść/wyjść

W tabeli błędów działania układów wejść/wyjść rejestrowane są takie błędy jak uszkodzenie modułów czy konflikty adresów.

Oprogramowanie może pracować w dowolnym trybie. Jeżeli oprogramowanie znajduje się w trybie **OFFLINE** informacje o błędach nie są wyświetlane. Są one natomiast pokazywane w trybach **ONLINE** i **MONITOR**. W trybie **ONLINE**, można usuwać błędy (funkcja ta może być zabezpieczona hasłem).

Po usunięciu informacji o błędzie pomimo, że błąd nadal występuje w systemie, nie jest on rejestrowany w tabeli.

Wyświetlanie dodatkowych informacji o błędzie

W tabeli błędów podane są podstawowe informacje o błędzie. Z poziomu oprogramowania można wyświetlić dodatkowe informacje o błędzie. Oprogramowanie umożliwia również uzyskanie opisu błędu w systemie heksadecymalnym.

Przy omawianiu błędów, w pozycji Usuwanie podano sposób usunięcia błędu. Przy niektórych opisach sposobu usuwania błędu podawana jest następująca informacja:

Display the PLC Fault Table on the Programmer. Contact GE Fanuc Field Service, giving them all the information contained in the fault entry. (Proszę wyświetlić tabelę błędów działania sterownika na programatorze. Skontaktować się z Działem Technicznym GE Fanuc i podać wszystkie informacje wyświetlane dla tego błędu).

Proszę zwrócić uwagę, że należy podać zarówno informacje bezpośrednio odczytane z tabeli błędów, jak i informacje zapisane w postaci liczb heksadecymalnych. Po dostarczeniu tych informacji, Dział Techniczny prześle instrukcje dalszego postępowania.

Część 2: Tabela błędów działania sterownika

Dla każdego błędu podano opis oraz sposób usuwania błędu. Niektóre błędy mogą być powodowane przez różne przyczyny. W takich przypadkach, kod błędu wyświetlany w dodatkowych informacjach o błędzie, umożliwia wskazanie przyczyny, która może powodować występowanie tego błędu. Kod ten, to dwie pierwsze cyfry heksadecymalne, w piątej kolumnie (proszę porównać z zamieszczonym poniżej przykładem).

01	000000	01030100	0902	0200	000000000000
				_____	Kod błędu (dwie pierwsze cyfry w piątej kolumnie cyfr)

Przyczyną występowania niektórych błędów może być uszkodzenie pamięci RAM sterownika. Błędy te mogą również wystąpić na skutek wyłączenia zasilania lub rozładowania baterii. W celu uniknięcia powtarzania tych samych instrukcji w przypadku, gdy powodem błędu działania może być uszkodzenie pamięci sterownika, podawana jest następująca, skrótowa instrukcja:

Skontrolować pamięć sterownika.

W takiej sytuacji należy:

1. Jeśli zasilanie systemu zostało wyłączone, wymienić baterię. Napięcie baterii może być bowiem niewystarczające do podtrzymywania zawartości pamięci.
2. Wymienić płytę jednostki centralnej. Przyczyną błędu może być awaria układów scalonych na tej płycie.

Zamieszczona poniżej tabela umożliwia szybkie odszukanie opis błędu w tym rozdziale. Każda z pozycji podawana jest w postaci, w jakiej wyświetlana jest na ekranie, w nawiasie podano polskie tłumaczenie komunikatu.

Błąd	Strona
Loss of, or Missing, Option Module (<i>Brak lub uszkodzenie modułu dodatkowego</i>)	3-8
Reset of, Addition of, or Extra, Option Module (<i>Ponowne uruchomienie, dodanie lub brak konfiguracji dla modułu</i>)	3-8
System Configuration Mismatch (<i>Niezgodność konfiguracji</i>)	3-9
Option Module Software Failure (<i>Błąd w oprogramowaniu wyspecjalizowanego modułu dodatkowego</i>)	3-10
Program Block Checksum Failure (<i>Błędna suma kontrolna programu sterującego</i>)	3-10
Low Battery Signal (<i>Wyczerpana bateria</i>)	3-10
Constant Sweep Time Exceeded (<i>Przekroczony czas trwania cyklu sterownika</i>)	3-11
Application Fault (<i>Błąd w działaniu programu sterującego</i>)	3-11
No User Program Present (<i>Brak programu sterującego</i>)	3-12
Corrupted User Program on Power-Up (<i>Wykrycie uszkodzenia programu sterującego podczas rozruchu</i>)	3-14
Password Access Failure (<i>Niepoprawne hasło dostępu do sterownika</i>)	3-12
PLC CPU System Software Failure (<i>Błąd w oprogramowaniu jednostki centralnej sterownika</i>)	3-13
Communications Failure During Store (<i>Awaria komunikacji podczas ładowania programu do sterownika</i>)	3-15

Wagi błędów i działania podejmowane w momencie ich wystąpienia

Błędy **krytyczne** powodują przejście sterownika w tryb zatrzymania (**STOP**), po zakończeniu cyklu, w którym błąd wystąpił. W przypadku wystąpienia błędów **diagnostycznych**, są one rejestrowane i ustawiane są odpowiednie zmienne systemowe. Błędy **informacyjne** są wyłącznie rejestrowane w tabeli błędów sterownika.

Loss of, or Missing, Option Module (Brak lub uszkodzenie modułu dodatkowego)

Błędy należące do tej grupy są rejestrowane w przypadku braku odpowiedzi z modułu programowalnego koprocatora PCM, modułu komunikacyjnego CMM lub modułu koprocatora wyświetlacza alfanumerycznego ADC. Błędy te mogą być zarejestrowane przy rozruchu sterownika, w przypadku stwierdzenia braku modułu lub, gdy moduł nie odpowiada na żądanie komunikacji. Są to błędy **diagnostyczne**.

Kod błędu:	1, 42
Nazwa:	Option Module Soft Reset Failed
Opis:	Jednostka centralna sterownika nie jest w stanie nawiązać komunikacji z jednym z dodatkowych, wyspecjalizowanych modułów.
Usuwanie:	<ol style="list-style-type: none"> (1) Spróbować ponownie uruchomić sterownik. (2) Wymienić moduł, który nie odpowiada na żądanie komunikacji. (3) Wyłączyć zasilanie systemu. Sprawdzić, czy moduł jest poprawnie zainstalowany w kasecie oraz czy wszystkie kable są poprawnie i trwale podłączone. (4) Wymienić kable przyłączeniowe.
Kod błędu:	Wszystkie pozostałe.
Nazwa:	Module Failure During Configuration
Opis:	Błąd rejestrowany w przypadku stwierdzenia uszkodzenia modułu w czasie rozruchu lub podczas zapisu konfiguracji.
Usuwanie:	<ol style="list-style-type: none"> (1) Wyłączyć zasilanie systemu. Wymienić moduł umieszczony w odpowiednim gnieździe kasyety sterownika.

Reset of, Addition of, or Extra, Option Module (Ponowne uruchomienie, dodanie, lub brak konfiguracji dla modułu)

Błędy z grupy **Reset of, Addition of, or Extra Option Module** są rejestrowane jeżeli moduł wyspecjalizowany (PCM, ADC, itp.) zostanie włączony, wyzerowany lub znaleziony w gnieździe, ale nie jest zdefiniowany w konfiguracji. Są to błędy **diagnostyczne**. Wraz z błędem podawane są dodatkowe informacje, bliżej charakteryzujące ten błąd.

Usuwanie:	<ol style="list-style-type: none"> (1) Uaktualnić konfigurację w celu włączenia do niej wyspecjalizowanego modułu. (2) Usunąć moduł z systemu.
------------------	--

System Configuration Mismatch (Błędna konfiguracja systemu)

Błędy z grupy **System Configuration Mismatch** są rejestrowane w przypadku stwierdzenia niezgodności pomiędzy modulem zainstalowanym w gnieździe, a modulem określonym w konfiguracji. Jest to błąd krytyczny.

Kod błędu:	1
Nazwa:	System Configuration Mismatch
Opis:	Oprogramowanie systemowe sterownika rejestruje ten błąd, jeżeli konfiguracja rzeczywista nie jest zgodna z konfiguracją zaprogramowaną.
Usuwanie:	Zidentyfikować niezgodność, a następnie ponownie skonfigurować moduł lub kasetę.
Kod błędu:	6
Nazwa:	System Configuration Mismatch
Opis:	Błąd ten wykazuje podobieństwo do błędu o kodzie nr 1, ponieważ występuje, jeżeli moduł zainstalowany w gnieździe nie jest zgodny z konfiguracją lub jeżeli skonfigurowany typ kasety nie jest zgodny z rzeczywiście zainstalowaną kasetą.
Usuwanie:	Zidentyfikować niezgodność, a następnie ponownie skonfigurować moduł lub kasetę.
Kod błędu:	18
Nazwa:	Unsupported Hardware
Opis:	W jednostce centralnej 311, 313, 323 lub w kasecie rozszerzającej zainstalowany jest moduł programowalnego koprocesora PCM lub inny moduł tego typu.
Usuwanie:	Usunąć moduł PCM lub moduł typu PCM, albo zainstalować jednostkę centralną obsługującą moduł PCM.
Kod błędu:	26
Nazwa:	Module busy—config not yet accept by module
Opis:	Nie można w danym momencie wprowadzić nowej konfiguracji dla modułu, ponieważ moduł ten zajęty jest przez inny proces.
Usuwanie:	Odczekać do momentu kiedy moduł zakończy realizowaną operację, po czym ponownie wprowadzić konfigurację.
Kod błędu:	51
Nazwa:	END Function Executed from SFC Action
Opis:	Wystąpienie tego błędu powodowane jest przez umieszczenie w programie sterującym SFC lub w programie wywoływanym z poziomu programu SFC, funkcji END.
Usuwanie:	Usunąć funkcję END z programu sterującego SFC lub z programu wywoływanego z poziomu programu SFC.

Option Module Software Failure (Błąd w oprogramowaniu wyspecjalizowanego modułu dodatkowego)

Powodem występowania błędów z grupy **Option Module Software Failure** jest wykrycie nieusuwalnego błędu w oprogramowaniu modułu programowalnego koprocatora PCM lub modułu koprocatora wizualizacji ADC. Jest to błąd **krytyczny**.

Kod błędu:	Wszystkie
Nazwa:	COMMREQ Frequency Too High
Opis:	Sygnaly żądania komunikacji są wysyłane do modułu z częstotliwością większą, niż zdolność modułu do ich przetwarzania.
Usuwanie:	Dokonać zmian w programie sterującym, aby sterownik wysyłał do modułu sygnały żądania komunikacji z mniejszą częstotliwością.

Program Block Checksum Failure (Błędna suma kontrolna programu)

Błędy z grupy **Program Block Checksum Failure** są rejestrowane, jeżeli jednostka centralna sterownika wykryje błędy w blokach programu sterującego przesłanego do sterownika. Są one również rejestrowane, jeżeli jednostka centralna sterownika wykryje błędną sumę kontrolną na etapie sprawdzania pamięci podczas rozruchu lub w trakcie pracy sterownika (w trybie **RUN**). Są to błędy **krytyczne**.

Kod błędu:	Wszystkie
Nazwa:	Program Block Checksum Failure
Opis:	Oprogramowanie systemowe sterownika rejestruje ten błąd, w przypadku stwierdzenia uszkodzenia bloku programu sterującego.
Usuwanie:	<ol style="list-style-type: none"> (1) Wyczyścić pamięć sterownika i ponownie przesłać program. (2) Wyświetlić tabelę błędów działania sterownika na ekranie komputera-programatora. Skontaktować się z serwisem firmy GE Fanuc podając wszystkie informacje dotyczące błędu.

Low Battery Signal (Rozładowanie baterii)

Błędy z grupy **Low Battery Signal** są rejestrowane w przypadku stwierdzenia zbyt niskiego napięcia baterii jednostki centralnej lub w przypadku zasygnalizowania przez jeden z modułów (np. moduł programowalnego koprocatora) niskiego napięcia baterii. Są to błędy **diagnostyczne**.

Kod błędu:	0
Nazwa:	Failed Battery Signal
Opis:	Bateria jednostki centralnej (lub innego modułu posiadającego baterię) jest wyczerpana.
Usuwanie:	Wymienić baterię. Nie wyłączać zasilania kasyty sterownika.
Kod błędu:	1
Nazwa:	Low Battery Signal
Opis:	Bateria jednostki centralnej (lub innego modułu posiadającego baterię) daje zbyt niskie napięcie.
Usuwanie:	Wymienić baterię. Nie wyłączać zasilania kasyty sterownika.

Constant Sweep Time Exceeded (Przekroczony czas trwania cyklu sterownika)

Błędy z grupy **Constant Sweep Time Exceeded** są rejestrowane, jeżeli jednostka centralna sterownika pracuje w trybie o stałym czasie trwania (**CONSTANT SWEEP**), a czas trwania cyklu przekroczył zdefiniowaną wartość. Dodatkowe dane zawierają rzeczywisty czas trwania cyklu (w pierwszych dwóch bajtach) i nazwę programu sterującego (w następnych ośmiu bajtach). Jest to błąd **diagnostyczny**.

- | | |
|------------------|--|
| Usuwanie: | (1) Zwiększyć czas trwania cyklu pracy sterownika. |
| | (2) Usunąć fragment programu sterującego. |

Application Fault (Błąd w działaniu programu sterującego)

Błędy z grupy **Application Fault** są rejestrowane, jeżeli jednostka centralna sterownika wykryje błąd w programie sterującym (błąd struktury programu). Są to błędy **diagnostyczne**, za wyjątkiem błędu Subroutine Call Stack Exceeded, który jest błędem **krytycznym**.

Kod błędu:	7
Nazwa:	Subroutine Call Stack Exceeded
Opis:	Podprogramy mogą być wywoływane z poziomu z innych podprogramów, dopuszczalne jest osiem poziomów zagnieżdżeń. Z poziomu jednego podprogramu można wywołać następny podprogram, która z kolei znowu może wywoływać następny podprogram, aż do dojścia do 8 poziomu.
Usuwanie:	Zmodyfikować program sterujący tak, aby nie przekraczać dopuszczalnej liczby zagnieżdżenia wywołań.
Kod błędu:	1B
Nazwa:	Comm Req Not Processed Due To PLC Memory Limitations
Opis:	Żądania bezzwłocznej komunikacji mogą być umieszczane w kolejce z większą prędkością niż są obsługiwane (np. jedno na jeden cykl). W przypadku takiej sytuacji, w momencie nagromadzenia się dużej liczby żądań powodującej spadek dostępnej pamięci poniżej dopuszczalnej, minimalnej wartości, żądania takie traktowane są jako błędy i nie są przetwarzane.
Usuwanie:	Zmniejszyć liczbę żądań komunikacji lub zredukować ilość danych wymienianych w systemie.
Kod błędu:	5A
Nazwa:	User Shut Down Requested
Opis:	Oprogramowanie systemowe sterownika (bloki funkcyjne) generują ten błąd informacyjny, gdy w programie sterującym zostanie wykonana funkcja SVCREQ #13 (powodująca zatrzymanie sterownika – przejście w tryb STOP).
Usuwanie:	Żadne działanie nie jest konieczne. Jest to jedynie błąd informacyjny.

No User Program Present (Brak programu sterującego)

Błędy z grupy **No User Program Present** są rejestrowane, jeżeli sterownik otrzyma instrukcję przejścia z trybu zatrzymania (**STOP**) do trybu pracy (**RUN**), a w pamięci sterownika nie ma programu sterującego. Jednostka centralna sprawdza obecność programu sterującego w momencie rozruchu. Jest to błąd **informacyjny**.

Usuwanie:	Załadować program sterujący przed przejściem do trybu RUN .
------------------	--

Corrupted User Program on Power-Up (Wykrycie uszkodzenia programu sterującego podczas rozruchu)

Błędy z grupy **Corrupted User Program on Power-Up** występują, jeżeli podczas rozruchu stwierdzone zostanie uszkodzenie pamięci RAM. Jednostka sterująca sterownika będzie pozostawać w trybie **STOP**, aż do momentu załadowania poprawnego programu sterującego i pliku konfiguracyjnego. Jest to błąd **krytyczny**.

Kod błędu:	1
Nazwa:	Corrupted User RAM on Power-Up
Opis:	Oprogramowanie systemowe sterownika rejestruje ten błąd w przypadku wykrycia w momencie rozruchu uszkodzenia pamięci RAM.
Usuwanie:	<ol style="list-style-type: none"> (1) Załadować ponownie program sterujący, plik konfiguracyjny oraz ewentualnie wartości zmiennych. (2) Wymienić baterię jednostki centralnej sterownika. (3) Wymienić płytę z rozszerzeniem pamięci jednostki centralnej sterownika. (4) Wymienić jednostkę centralną sterownika.
Kod błędu:	2
Nazwa:	Illegal Boolean OpCode Detected
Opis:	Oprogramowanie systemowe sterownika rejestruje ten błąd przy wykryciu niewłaściwych instrukcji w programie sterującym.
Usuwanie:	<ol style="list-style-type: none"> (1) Załadować ponownie program sterujący oraz ewentualne wartości zmiennych. (2) Wymienić płytę z rozszerzeniem pamięci jednostki centralnej sterownika. (3) Wymienić jednostkę centralną sterownika.

Password Access Failure (Niepoprawne hasło dostępu sterownika)

Błędy z grupy **Password Access Failure** są rejestrowane w przypadku wprowadzenia niewłaściwego hasła dostępu do jednego z poziomów dostępu do sterownika. Jest to błąd **informacyjny**.

Usuwanie:	Ponowić próbę używając poprawnego hasła dostępu.
------------------	--

PLC CPU System Software Failure (Błąd w oprogramowaniu jednostki centralnej)

Błędy z grupy **PLC CPU System Software Failure** są rejestrowane wskutek wielu różnych przyczyn związanych z działaniem oprogramowania systemowego sterowników 90-30, 90-20 i Micro. Błędy te mogą występować w różnych momentach pracy sytemu. Są to błędy **krytyczne**, powodujące **natychmiastowe** przejście jednostki centralnej sterownika w tryb pracy uniemożliwiający wykonanie jakiegokolwiek operacji (**ERROR-SWEEP**), oprócz komunikacji z komputerem-programatorem. Jedynym sposobem anulowania tego stanu jest wyłączenie i ponowne włączenie zasilania sterownika. Jest to błąd **krytyczny**.

Kod błędu:	1 do B
Nazwa:	User Memory Could Not Be Allocated
Opis:	Oprogramowanie systemowe zarządzające pamięcią sterownika rejestruje ten błąd, w przypadku próby przydzielenia niedopuszczalnych bloków pamięci RAM. Takie błędy nie powinny występować w systemie.
Usuwanie:	Wyświetlić tabelę błędów działania sterownika na ekranie komputera-programatora. Skontaktować się z serwisem firmy GE Fanuc, podając wszystkie informacje dotyczące błędu.
Kod błędu:	D
Nazwa:	System Memory Unavailable
Opis:	Oprogramowanie systemowe rejestruje ten błąd, gdy program zarządzający pamięcią odmawia przydzielenia pamięci systemowej z powodu braku miejsca na stosie systemowym. Jeśli błąd pojawia się w czasie wykonywania funkcji DO I/O, jest to jedynie błąd <i>informacyjny</i> . Jeżeli natomiast zostanie zarejestrowany podczas rozruchu sterownika lub automatycznego konfigurowania jest to błąd <i>krytyczny</i> .
Usuwanie:	Wyświetlić tabelę błędów działania sterownika na ekranie komputera-programatora. Skontaktować się z serwisem firmy GE Fanuc, podając wszystkie informacje dotyczące błędu.
Kod błędu:	E
Nazwa:	System Memory Could Not Be Freed
Opis:	Oprogramowanie systemowe rejestruje ten błąd, jeżeli program zarządzający pamięcią odmawia zwolnienia bloku pamięci systemowej. Błąd taki może pojawić się jedynie w czasie wykonywania funkcji DO I/O.
Usuwanie:	(1) Wyświetlić tabelę błędów działania sterownika na ekranie komputera-programatora. Skontaktować się z serwisem firmy GE Fanuc podając wszystkie informacje dotyczące błędu. (2) Skontrolować pamięć sterownika.
Kod błędu:	10
Nazwa:	Invalid Scan Request of the I/O Scanner
Opis:	Oprogramowanie systemowe rejestruje ten błąd, jeżeli system operacyjny lub funkcja DO I/O nie żądają ani pełnego, ani częściowego skanowania wejść i wyjść. Błąd ten nie powinien występować.
Usuwanie:	Wyświetlić tabelę błędów działania sterownika na ekranie komputera-programatora. Skontaktować się z serwisem firmy GE Fanuc, podając wszystkie informacje dotyczące błędu.
Kod błędu:	13
Nazwa:	PLC Operating Software Error
Opis:	Błąd ten dotyczy pewnych problemów w funkcjonowaniu oprogramowania sterownika. Takie błędy nie powinny występować w systemie.
Usuwanie:	(1) Wyświetlić tabelę błędów działania sterownika na ekranie komputera-programatora. Skontaktować się z serwisem firmy GE Fanuc podając wszystkie informacje dotyczące błędu. (2) Skontrolować pamięć sterownika.

Kod błędu:	14, 27
Nazwa:	Corrupted PLC Program Memory
Opis:	Oprogramowanie sterownika rejestruje ten błąd, w przypadku wystąpienia pewnych problemów związanych z funkcjonowaniem pamięci. Błędy te nie powinny występować w systemie.
Usuwanie:	(1) Wyświetlić tabelę błędów działania sterownika na ekranie komputera-programatora. Skontaktować się z serwisem firmy GE Fanuc podając wszystkie informacje dotyczące błędu. (2) Skontrolować pamięć sterownika.
Kod błędu:	27 do 4E
Nazwa:	PLC Operating Software Error
Opis:	Oprogramowanie sterownika rejestruje ten błąd, w przypadku wystąpienia pewnych problemów związanych z funkcjonowaniem pamięci. Błędy te nie powinny występować w systemie.
Usuwanie:	Wyświetlić tabelę błędów działania sterownika na ekranie komputera-programatora. Skontaktować się z serwisem firmy GE Fanuc, podając wszystkie informacje dotyczące błędu.
Kod błędu:	4F
Nazwa:	Communications Failed
Opis:	Oprogramowanie sterownika rejestruje ten błąd, jeżeli na wysłane żądanie otrzymuje odmowę nawiązania komunikacji.
Usuwanie:	(1) Sprawdzić, czy szyna komunikacyjna funkcjonuje poprawnie. (2) Wymienić wyspecjalizowany moduł dodatkowy, który nie odpowiada na żądanie komunikacji.
Kod błędu:	50, 51, 53
Nazwa:	System Memory Errors
Opis:	Oprogramowanie sterownika rejestruje ten błąd, gdy program zarządzający pamięcią nie jest w stanie przydzielić bloku pamięci systemowej z powodu braku wolnej pamięci lub jej uszkodzenia.
Usuwanie:	(1) Wyświetlić tabelę błędów działania sterownika na ekranie komputera-programatora. Skontaktować się z serwisem firmy GE Fanuc podając wszystkie informacje dotyczące błędu. (2) Skontrolować pamięć sterownika.
Kod błędu:	52
Nazwa:	Backplane Communications Failed
Opis:	Oprogramowanie sterownika (proces obsługi żądań komunikacji) rejestruje ten błąd, gdy nie jest w stanie nawiązać komunikacji przez szynę komunikacyjną.
Usuwanie:	(1) Sprawdzić, czy szyna komunikacyjna funkcjonuje poprawnie. (2) Wymienić wyspecjalizowany moduł dodatkowy, który nie odpowiada na żądanie komunikacji. (3) Sprawdzić, czy kabel równoległy programatora jest prawidłowo podłączony.
Kod błędu:	Wszystkie pozostałe.
Nazwa:	PLC CPU Internal System Error
Opis:	Wystąpienie wewnętrznego błędu systemowego, który nie powinien pojawić się w systemie.
Usuwanie:	Wyświetlić tabelę błędów działania sterownika na ekranie komputera-programatora. Skontaktować się z serwisem firmy GE Fanuc, podając wszystkie informacje dotyczące błędu.

Communications Failure During Store (Awaria komunikacji podczas ładowania programu do sterownika)

Błędy z grupy **Communications Failure During Store** są rejestrowane podczas procesu przesyłania bloków programu sterującego i innych danych do pamięci sterownika. Proces ten jest rozpoczynany i kończony specjalnymi instrukcjami. Błąd ten jest rejestrowany w przypadku przerwania lub wystąpienia innych zakłóceń w komunikacji w czasie przesyłania. Tak długo, jak błąd ten obecny jest w systemie, sterownik nie może zostać uruchomiony (nie może przejść w tryb **RUN**).

Błąd ten nie jest automatycznie usuwany przy rozruchu sterownika; użytkownik musi zrobić to samodzielnie. Jest to błąd **krytyczny**.

Usuwanie:	Usunąć komunikat o błędzie i spróbować ponownie załadować program lub plik konfiguracyjny.
------------------	--

Część 3: Tabela błędów działania układów wejść/wyjść

Dane o błędach zawarte w tabeli błędów działania układów wejść/wyjść mogą być podzielone na trzy grupy:

- Kategoria błędu
- Typ błędu
- Opis błędu

W podanych poniżej opisach podano kategorię błędów, ale nie podano typu i grupy błędów.

Dla każdego błędu podano opis oraz sposób usuwania błędu. Niektóre błędy mogą być powodowane przez różne przyczyny. W takich przypadkach, po wciśnięciu klawiszy CTRL-F wyświetlany jest kod błędu wraz z dodatkowymi informacjami, co umożliwi wskazanie przyczyny, która spowodowała wystąpienie tego błędu. (Więcej informacji na temat korzystania ze skrótu klawiaturowego CTRL-F podano w Załączniku B "Interpretacja tabeli błędów" niniejszego podręcznika.) Kategoria błędu to dwie pierwsze cyfry heksadecymalne, w piątej kolumnie numerów (proszę porównać z zamieszczonym poniżej przykładem).

<pre>02 1F0100 00030101FF7F 0302 0200 84000000000003 _____ Kategorie błędu (pierwsze dwie cyfry heksadecymalne w piątej kolumnie)</pre>

Zamieszczona poniżej tabela umożliwia szybkie odszukanie opisu błędu. Każda z pozycji podawana jest w postaci, w jakiej wyświetlana jest na ekranie.

Loss of I/O Module (Brak zadeklaowanego modułu wejść/wyjść)

Kategoria **Loss of I/O Module** dotyczy modułów wejść/wyjść analogowych i cyfrowych sterowników 90-30. Dla kategorii tej nie są podawane typy błędów i opisy błędów. Są to błędy **diagnostyczne**.

Opis:	Oprogramowanie sterownika rejestruje ten błąd, jeżeli moduł wejść/wyjść nie reaguje na rozkazy wysyłane przez jednostkę centralną lub gdy plik konfiguracyjny wykazuje obecność w jednym z gniazd kasety sterownika modułu, którego fizycznie nie ma.
Usuwanie:	<ol style="list-style-type: none"> (1) Wymienić moduł. (2) Uaktualnić plik konfiguracyjny. (3) Wyświetlić tabelę błędów działania sterownika na ekranie komputera programatora. Skontaktować się z serwisem firmy GE Fanuc podając wszystkie informacje dotyczące błędu.

Addition of I/O Module (Dołączony niezadeklarowany moduł wejść/wyjść)

Kategoria **Addition of I/O Module** dotyczy modułów wejść/wyjść analogowych i cyfrowych sterowników serii 90-30. Dla kategorii tej nie są podawane typy błędów i opisy błędów. Są to błędy **diagnostyczne**.

Opis:	Oprogramowanie sterownika rejestruje ten błąd, w przypadku wznowienia pracy przez moduł wejść/wyjść, dla którego wcześniej stwierdzono uszkodzenie.
Usuwanie:	<ol style="list-style-type: none"> (1) Jeżeli moduł został usunięty lub wymieniony, nie jest potrzebne jakiegokolwiek działanie. (2) Uaktualnić dane konfiguracyjne w celu uwzględnienia dodatkowego modułu lub usunąć moduł z systemu.
Opis:	Oprogramowanie sterownika rejestruje ten błąd, gdy wykryta zostaje fizyczna obecność modułu wejść/wyjść sterowników serii 90-30, który nie został zadeklarowany w pliku konfiguracyjnym sterownika.
Usuwanie:	<ol style="list-style-type: none"> (1) Wymienić moduł (może być zainstalowany w niewłaściwym gnieździe). (2) Uaktualnić dane konfiguracyjne w celu uwzględnienia dodatkowego modułu.

Rozdział 4

Styki i przekaźniki

Niniejszy rozdział zawiera opis styków, przekaźników i szyn, wykorzystywanych w programach sterujących w języku drabinkowym.

Styk/ przekaźnik	Strona
Przekaźniki o stykach otwartych i przekaźniki o stykach zamkniętych	4-2
Styki otwarte i styki zamknięte	4-3
Przekaźniki o stykach otwartych i przekaźniki o stykach zamkniętych z pamięcią	4-4
Przekaźniki uaktywniane zboczem narastającym sygnału i przekaźniki uaktywniane zboczem opadającym sygnału	4-5
Przekaźniki ustawialne SET i RESET	4-6
Przekaźniki ustawialne SET i RESET z pamięcią	4-7
Szyny poziome i pionowe	4-7
Przekaźniki i styki kontynuacji	4-8

Styki

Styki są wykorzystywane do sterowania przepływem sygnału. Od stanu zmiennej przypisanej do styku oraz od typu styku zależy, czy będzie on przewodził sygnał. Zmienna jest ustawiona (ON) jeżeli jej wartość jest równa 1, zmienna jest wyzerowana (OFF) jeżeli jej wartość jest równa 0.

Tabela 4-1. Typy styków

Typ	Wyświetlacz	Dopływ sygnału do prawej strony
Przekaźnik o stykach otwartych	— —	Jeżeli zmienna ma wartość 1
Styk zamknięty	—/ —	Jeżeli zmienna ma wartość 0
Styk kontynuacji	<+>——	Jeżeli zmienna związana z poprzednim stykiem kontynuacji ma wartość 1.

Przełączniki

Przełączniki są wykorzystywane do ustawiania wartości zmiennych dyskretnych. Dopływ sygnału do przełącznika musi być sterowany przez inne elementy logiczne. Przełączniki natychmiast zmieniają wartość zmiennych, nie przesyłają one sygnału do swojej prawej strony. Jeżeli wartość zmiennej przypisanej do przełącznika ma decydować o wykonaniu pewnej części programu sterującego, należy tam zastosować zmienną wewnętrzną lub przełącznik i styk kontynuacji.

Przełączniki są zawsze umieszczane skrajnie, po prawej stronie szczelby programu sterującego. Szczelby może zawierać do ośmiu przełączników.

Typ przełącznika należy dobrać stosownie do żądanego działania programu sterującego. Stan przełączników z pamięcią jest zapamiętywany po wyłączeniu zasilania lub po przejściu sterownika z trybu zatrzymania (**STOP**) do trybu pracy (**RUN**). Stan przełączników bez pamięci jest ustawiany na zero po wyłączeniu zasilania lub po przejściu sterownika z trybu zatrzymania (**STOP**) do trybu pracy (**RUN**).

Tabela 4-2. Typy przełączników

Przełącznik	Symbol	Stan doprowadz. sygnału	Działanie
Przełącznik o stykach otwartych	—(O)—	ON	Ustawia wartość zmiennej na 1
		OFF	Ustawia wartość zmiennej na 0.
Przełącznik o stykach zamkniętych	—(/)—	ON	Ustawia wartość zmiennej na 0.
		OFF	Ustawia wartość zmiennej na 1
Przełącznik o stykach otwartych z pamięcią	—(M)—	ON	Ustawia wartość zmiennej na 1 i zapamiętuje tę wartość.
		OFF	Ustawia wartość zmiennej na 0 i zapamiętuje tę wartość.
Przełącznik o stykach zamkniętych z pamięcią	—(/M)—	ON	Ustawia wartość zmiennej na 0 i zapamiętuje tę wartość.
		OFF	Ustawia wartość zmiennej na 1 i zapamiętuje tę wartość.
Przełącznik uaktywniany zboczem narastającym sygnału	—(↑)—	OFF→ON	Jeżeli zmienna ma wartość 1, jej wartość jest zmieniana na 0 na okres jednego cyklu.
Przełącznik uaktywniany zboczem opadającym sygnału	—(↓)—	ON→OFF	Jeżeli zmienna ma wartość 1, jej wartość jest zmieniana na 0 na okres jednego cyklu.
Przełącznik SET	—(S)—	ON	Ustawia zmienną na 1, aż do momentu jej wyzerowania poprzez —(R)—.
		OFF	Nie zmieniać statusu przełącznika.
Przełącznik RESET	—(R)—	ON	Ustawia zmienną na 0, aż do momentu zmiany jej wartości na 1 poprzez przełącznik —(S)—.
		OFF	Nie zmieniać statusu przełącznika.
Przełącznik SET z pamięcią	—(SM)—	ON	Ustawia zmienną na 1, aż do momentu jej wyzerowania poprzez —(RM)—.
		OFF	Nie zmieniać statusu przełącznika.
Przełącznik RESET z pamięcią	—(RM)—	ON	Ustawia zmienną na 0, aż do momentu zmiany jej wartości na 1 poprzez przełącznik —(SM)—.
		OFF	Nie zmieniać statusu przełącznika.
Przełącznik kontynuacji	—<+>	ON	Zwarcie następującego po nim styku kontynuacji.
		OFF	Rozwarcie następującego po nim styku kontynuacji.

Styk otwarty—| |—

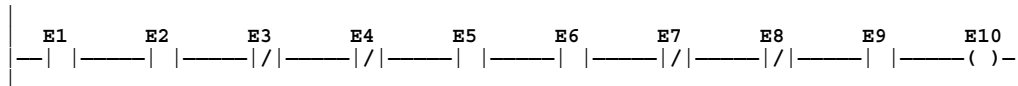
Styk taki działa jak wyłącznik, który przewodzi sygnał (zawiera styki), gdy wartość logiczna przypisanej do niego zmiennej wynosi 1.

Styk zamknięty —|/|—

Styk taki działa jak wyłącznik, którego styki pozostają zwarte (przewodzi sygnał), gdy wartość logiczna przypisanej do niego zmiennej wynosi 0.

Przykład

Poniżej pokazano przykładowy szczebel z 10 elementami, o nazwach pomocniczych od E1 do E10. Przekaznik E10 ma wartość logiczną 1, jeżeli zmienne E1, E2, E3, E5, E6 i E9 są równe 1, a zmienne E3, E4, E7 i E8 są równe 0.

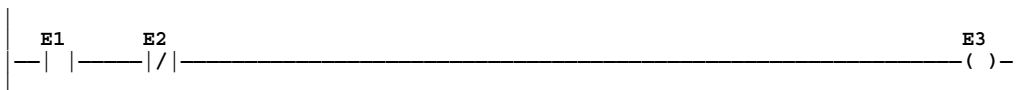


Przekaznik o stykach otwartych —()—

Przekaznik ten ustawia wartość przypisanej do niego zmiennej na jeden po doprowadzeniu do niego sygnału. Jest to przekaznik bez pamięci, nie może on być zatem zastosowany wraz ze zmiennymi z pamięcią stanu (%SA, %SB, %S.C.) lub %G.

Przykład

W zamieszczonym poniżej przykładzie, przekaznik ustawia wartość przypisanej do niego zmiennej na 1, jeżeli wartość E1 jest równa 1, a wartość E2 jest równa 0.

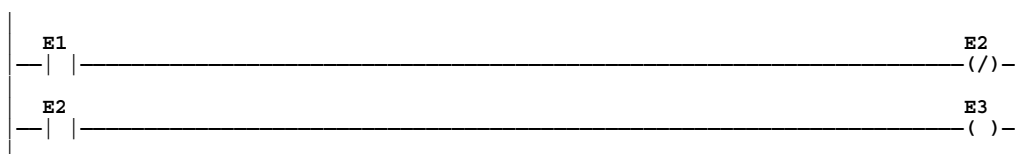


Przełącznik o stykach zamkniętych —(I)—

Przełącznik ten ustawia wartość przypisaną do niego zmiennej dyskretnej na jeden po doprowadzeniu do niego sygnału. Jest to przełącznik bez pamięci, nie może on być zatem stosowany ze zmiennymi z pamięcią stanu (%SA, %SB, %SC lub %G).

Przykład

W zamieszczonym poniżej przykładzie, przełącznik E3 ustawia wartość przypisaną do niego zmiennej na 1, jeżeli wartość E1 jest równa 0.



Przełącznik o stykach otwartych z pamięcią —(M)—

Podobnie jak zwykły przełącznik o stykach otwartych, przełącznik ten ustawia wartość przypisaną do niego zmiennej na jeden, po doprowadzeniu sygnału wejściowego. Stan przełącznika zostaje podtrzymany w przypadku zaniku zasilania sterownika. Z tego powodu, nie może być on zatem stosowany ze zmiennymi bez pamięci stanu (%T)

Przełącznik o stykach zamkniętych z pamięcią —(/M)—

Przełącznik ten ustawia wartość przypisaną do niego zmiennej dyskretnej na jeden, gdy nie dopływa do niego sygnał. Stan przełącznika zostaje podtrzymany w przypadku zaniku zasilania sterownika. Z tego powodu, nie może być on zatem zastosowany wraz ze zmiennymi, które nie posiadają pamięci stanu (%T)

Przełącznik uaktywniany zboczem narastającym sygnału —(↑)—

Jeżeli zmienna przypisana do przełącznika uaktywnianego zboczem narastającym sygnału ma wartość 0, doprowadzenie sygnału do przełącznika powoduje ustawienie wartości tej zmiennej na 1. Styki powiązane z takim przełącznikiem zmieniają swój stan na okres jednego cyklu pracy sterownika. (Jeżeli szczybel zawierający przełącznik zostanie pominięty w czasie kolejnych cykli, wartość tej zmiennej jest nadal równa 1, przez cały czas trwania tych cykli). Przełącznik ten może zostać wykorzystany do ustalania wartości zmiennej na okres jednego cyklu.

Zmienna powinna być wykorzystywana w programie sterującym jako przełącznik uaktywniany zboczem sygnału tylko jeden raz, co pozwoli na skorzystanie z charakterystycznego dla tego przełącznika ustawiania wartości na okres jednego cyklu.

Przełączniki ustawiane zboczem sygnału mogą być stosowane ze zmiennymi z pamięcią stanu lub zmiennymi bez pamięci stanu (tzn. %Q, %M, %T, %G, %SA, %SB lub %SC).

Przełącznik uaktywniany zboczem opadającym sygnału —(↓)—

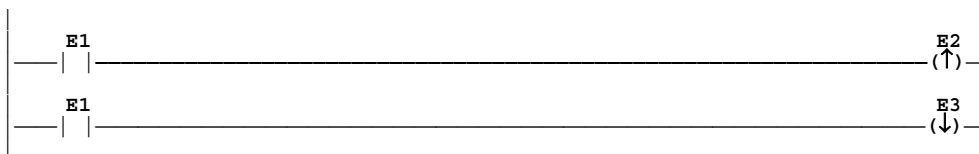
Jeżeli zmienna przypisana do tego przełącznika ma wartość 0, po zaprzestaniu doprowadzania sygnału do przełącznika, zmienna jest ustawiana na 1, a styki połączone z tym przełącznikiem zmieniają stan na okres jednego cyklu pracy sterownika.

Każda zmienna powinna być wykorzystywana w programie sterującym jako przełącznik uaktywniany zboczem sygnału tylko jeden raz, co pozwoli na skorzystanie z charakterystycznego dla tego przełącznika ustawiania wartości na okres jednego cyklu.

Przełączniki ustawiane zboczem sygnału mogą być stosowane ze zmiennymi z pamięcią stanu lub zmiennymi bez pamięci stanu (tzn. %Q, %M, %T, %G, %SA, %SB lub %SC).

Przykład

W zamieszczonym poniżej przykładzie, zmiana wartości zmiennej E1 z 0 na 1 powoduje doprowadzenie sygnału do przełączników E2 i E3, na skutek czego zmienna związana z przełącznikiem E2 przyjmuje wartość 1 na okres jednego cyklu. Jeżeli zmienna E1 zmieni wartość z 1 na 0, sygnał przestaje dopływać do E2 i E3, przez co zmienna związana z przełącznikiem E3 przyjmuje wartość 1 na okres jednego cyklu.



Przełącznik ustawialny SET —(S)—

Przełączniki SET i RESET są przełącznikami bez pamięci, które można zastosować do ustawiania na pewien czas wartości przypisanej do nich, tej samej zmiennej. Gdy do przełącznika SET dopływa sygnał, wartość przypisanej do niego zmiennej zostaje ustawiona na 1 i jest utrzymywana do momentu, aż sygnał dopłynie do przełącznika RESET do którego przypisano tę samą zmienną, niezależnie od tego, czy przez ten czas nadal dopływa sygnał do przełącznika SET, czy nie.

Przełącznik SET ustawia przypadkową wartość bitu chwilowego przełączenia na wartość przeciwną danej zmiennej. (Proszę porównać z informacjami podanymi w "Bity zmiany stanu i wymuszania wartości" w Rozdziale 2 "Działanie systemu".)

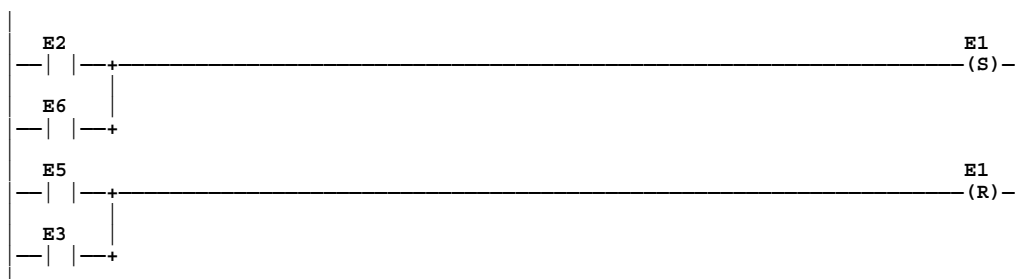
Przełącznik ustawialny RESET —(R)—

Gdy sygnał dopływa do przełącznika RESET, wartość zmiennej jest ustawiana na 0. Zmienna ma wartość 0, aż do momentu zmiany jej wartości przez inny, przypisany do niej przełącznik. Jej wartość jest zmieniana stosownie do przełącznika, do którego dopływa sygnał.

Przełącznik RESET ustawia przypadkową wartość bitu chwilowego przełączenia na wartość przeciwną danej zmiennej. (Proszę porównać z informacjami podanymi w "Bity zmiany stanu i wymuszania wartości" w Rozdziale 2 "Działanie systemu".)

Przykład

W zamieszczonym poniżej przykładzie, zmienna E1 przypisana do przekaźnika ma wartość 1 zawsze, ilekroć zmienna E2 lub E6 jest równa 1. Wartość zmiennej przypisanej do E1 jest zmieniana na 0 zawsze, ilekroć zmienna E5 lub E3 jest równa 1.



Uwaga

Jeżeli kontrola wielokrotnego wykorzystania zmiennych w przekaźnikach ustawiona jest na SINGLE, zmienne %M i %Q mogą być przypisane wyłącznie do jednego przekaźnika, nie mniej jednak, ta sama zmienna tego typu może być jednocześnie przypisana do przekaźnika SET i RESET. Jeżeli kontrola wielokrotnego wykorzystania zmiennych w przekaźnikach jest ustawiona na WARN MULTIPLE lub MULTIPLE, każda ze zmiennych może być używana z wieloma normalnymi przekaźnikami lub przekaźnikami Set i przekaźnikami RESET. W przypadku wielokrotnego wykorzystywania, wartość zmiennej można ustawić na 1 przy pomocy zarówno przekaźnika SET jak i normalnego przekaźnika, a następnie zmienić jej wartość na 0 przy pomocy przekaźnika RESET lub normalnego przekaźnika.

Przekaźnik SET z pamięcią —(SM)—

Przekaźniki SET i RESET z pamięcią są podobne w działaniu do przekaźników SET i RESET bez pamięci, lecz ich stan jest podtrzymywany w przypadku wyłączenia zasilania sterownika lub po przejściu sterownika z trybu **STOP** do trybu **RUN**. Po doprowadzeniu sygnału do przekaźnika SET z pamięcią, wartość związanej z nim zmiennej jest ustawiana na 1. Wartość ta pozostaje równa 1 do momentu wyzerowania za pomocą przekaźnika RESET z pamięcią.

Przekaźnik SET z pamięcią ustawia przypadkową wartość bitu zmiany stanu. (Proszę porównać z informacjami podanymi w "Bity zmiany stanu i wymuszania wartości" w Rozdziale 2 "Działanie systemu".)

Przekaźnik RESET z pamięcią —(RM)—

Przekaźnik ten ustawia wartość przypisanej do niego zmiennej na zero, po doprowadzeniu sygnału wejściowego. Wartość ta pozostaje równa 0, do momentu jej zmiany za pomocą przekaźnika SET z pamięcią. Stan tego przekaźnika jest zapamiętywany po wyłączeniu zasilania lub po przejściu sterownika z trybu zatrzymania (**STOP**) do trybu pracy (**RUN**).

Przekaźnik RESET z pamięcią ustawia przypadkową wartość bitu zmiany stanu. (Proszę porównać z informacjami podanymi w "Bity zmiany stanu i wymuszania wartości" w Rozdziale 2 "Działanie systemu".)

Szyny

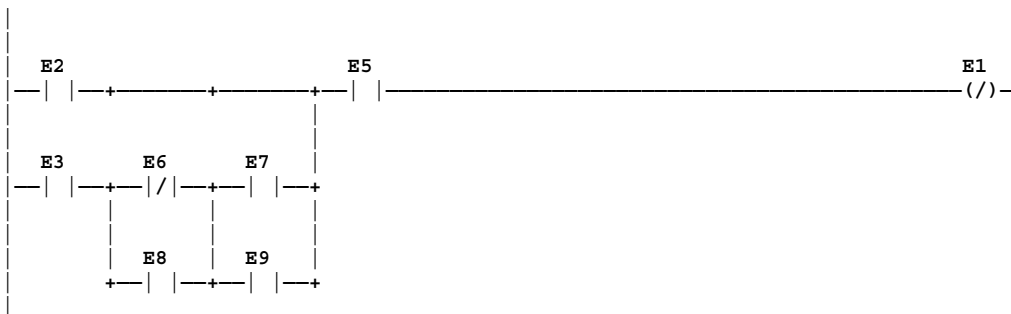
Szyny poziome i pionowe służą do łączenia elementów logicznych programu sterującego. Umożliwiają przepływ sygnału sterującego z lewej do prawej strony drabiny logicznej programu sterującego.

Uwaga

Za pomocą szyny poziomej nie można połączyć funkcji lub przekaźnika z lewą stroną sygnału. Jeżeli funkcja ma być wywoływana w każdym cyklu, można skorzystać ze zmiennej %S7, ALW_ON (zawsze włączona) powiązanej ze stykiem otwartym.

Przykład

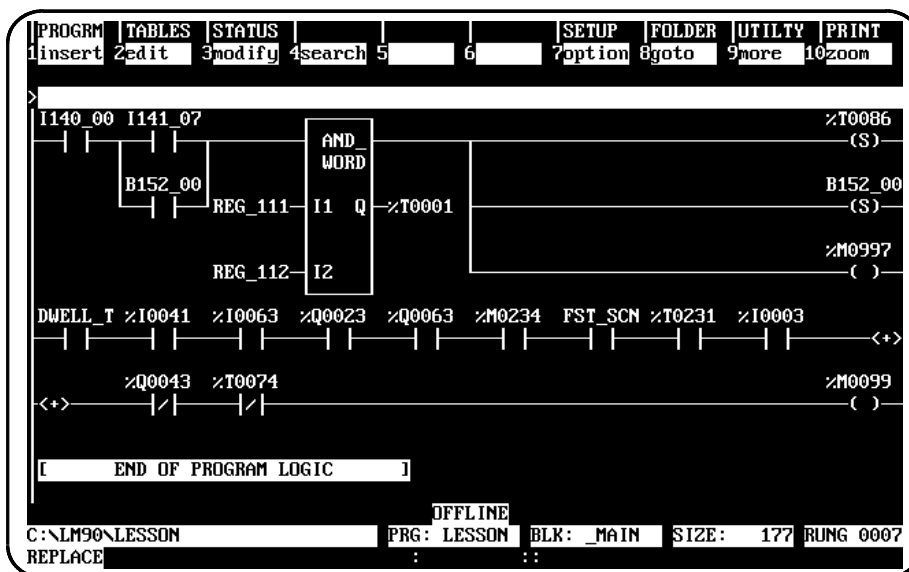
W zamieszczonym poniżej przykładzie, styki E2 i E5 połączone są za pomocą szyn poziomych. Szyny pionowe wykorzystane są do połączenia styków E3, E6, E7, E8 i E9 do E2.



Przełączniki kontynuacji (—<+>) i styki (<+>—) kontynuacji

Przełączniki kontynuacji (—<+>) i styki kontynuacji (<+>—) służą do przedłużenia szczebla drabiny logicznej programu sterującego poza limit dziesięciu kolumn. Stan ostatniego przełącznika kontynuacji jest automatycznie przenoszony na następujący po nim styk. Warunkiem wykonania części logicznej styku kontynuacji jest umieszczenie przed nim przełącznika kontynuacji. Stan styku kontynuacji jest zerowany w przypadku przejścia sterownika z trybu **STOP** do trybu **RUN**, a sygnał nie będzie przepływał aż do momentu kiedy wartość zmiennej związanej z przełącznikiem uaktywnianym z boczem sygnału zostanie ustawiona na 1, od momentu przejścia w tryb **RUN**.

W szczeblu może wystąpić tylko jeden przełącznik kontynuacji; styk kontynuacji można umieścić wyłącznie w pierwszej kolumnie szczebla, a przełącznik kontynuacji musi być umieszczony w 10 kolumnie szczebla. Poniżej pokazano przykład zastosowania przełącznika i styku kontynuacji.



Rozdział 5

Liczniki i przekaźniki czasowe

W rozdziale tym opisano przekaźniki czasowe oraz liczniki dodające i odejmujące. Dane związane z tymi blokami funkcyjnymi są pamiętane w przypadku wyłączenia zasilania.

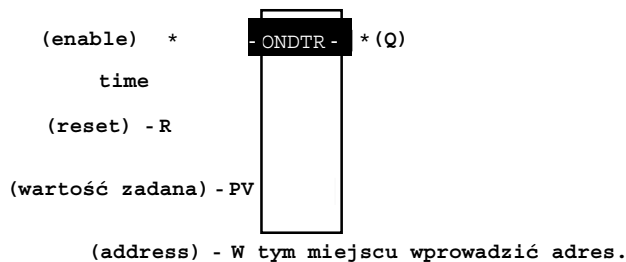
Oznaczenie skrótowe	Element	Strona
ONDTR	Przełącznik czasowy załączający z pamięcią	5-3
TMR	Przełącznik czasowy załączający	5-5
OFDT	Przełącznik czasowy wyłączający	5-8
UPCTR	Licznik dodający	5-11
DNCTR	Licznik odejmujący	5-12

Blok danych sterujących przekaźników czasowych i liczników

Każdy przekaźnik czasowy i licznik wykorzystuje trzy słowa (rejstry) pamięci typu %R, do zapamiętywania następujących parametrów:

wartość bieżąca (CV)	słowo 1
wartość zadana (PV)	słowo 2
słowo sterujące	słowo 3

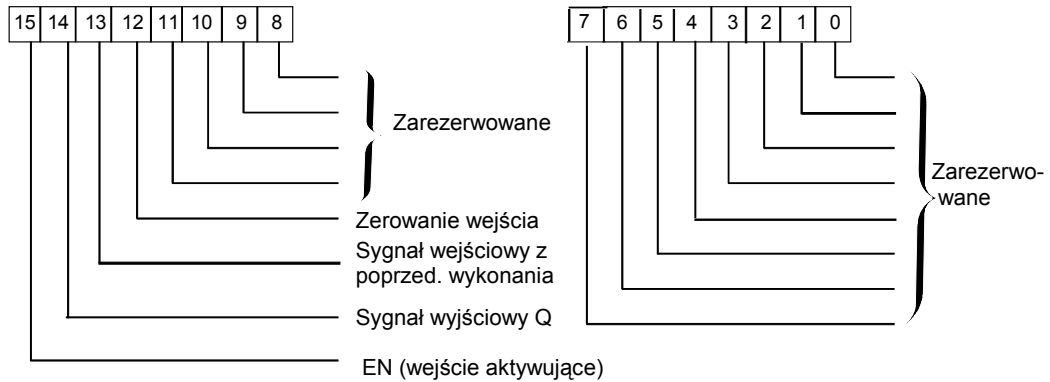
W bloku funkcyjnym przekaźnika czasowego lub licznika należy podać adres początkowy dla tych trzech słów (rejestrów), bezpośredni pod symbolem graficznym wykorzystywanego bloku. Przykładowo:



Uwaga

Nie należy podawać sąsiadujących ze sobą rejestrów jako adresów bloków danych sterujących przekaźników czasowych/liczników. Oprogramowanie Logicmaster *nie* sprawdza, ani też nie ostrzega w przypadku pokrywania się bloków danych sterujących. Przekaźniki czasowe i liczniki nie będą poprawnie pracować, jeżeli wartość bieżąca jednego bloku danych sterujących zostanie umieszczona w miejscu zajmowanym przez wartość zadana, wykorzystywaną przez inny blok danych sterujących.

Słowo sterujące zapamiętuje stan wejść i wyjść cyfrowych, powiązanych z danym blokiem funkcyjnym, zgodnie z przedstawionym poniżej formatem:



Bity 0 do 11 wykorzystywane są do zapewnienia przekaźnikowi czasowemu dokładności, nie są one wykorzystywane w przypadku liczników.

Uwaga

Należy zachować ostrożność, jeżeli ten sam adres wykorzystywany jest dla wartości zadanej PV jak i dla drugiego słowa w bloku danych sterujących. Jeżeli wartość zadana PV nie jest stałą, wartość PV jest zwykle ustawiana w innym miejscu niż to drugie słowo. Niektóre programy sterujące korzystają z adresu drugiego słowa w celu uzyskania wartości PV, jak na przykład wywołanie %R0102, jeżeli blok danych rozpoczyna się od %R101. Pozwala to na zmianę wartości PV w czasie pracy przekaźnika czasowego czy licznika. Programy sterujące mogą odczytywać pierwsze słowo zawierające wartość bieżącą CV i trzecie Słowo sterujące, ale nie mogą zmieniać ich wartości, ponieważ zakłóci to pracę bloku funkcyjnego.

Uwaga odnośnie niektórych operacji bitowych

W przypadku korzystania z funkcji BTST, BSET, BCLR i BPOS, bity numerowane są od 1 do 16, a NIE od 0 do 15, jak w przykładzie powyżej.

Przełącznik czasowy załączający z pamięcią (ONDTR)

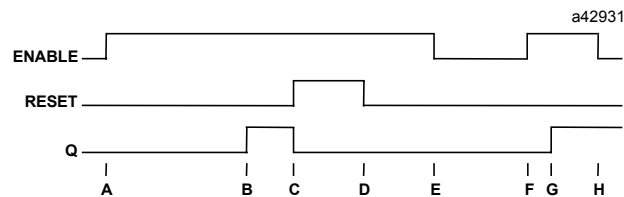
Przełącznik czasowy z pamięcią zlicza czas, gdy dopływa do niego sygnał i zatrzymuje naliczoną wartość, gdy sygnał przestaje dopływać. Czas może być zliczany w dziesiątych, setnych lub tysięcznych częściach sekundy. Zakres mierzonej wartości może wynosić od 0 do +32767 jednostek czasu. Wartość bieżąca przełącznika jest przechowywana w przypadku awarii zasilania sterownika; nie jest przeprowadzana automatyczna inicjalizacja po włączeniu zasilania.

Po pierwszym doprowadzeniu sygnału do przełącznika, rozpoczyna on naliczanie czasu (wartość bieżąca). Po napotkaniu przełącznika czasowego w drabinie programu sterującego, wartość bieżąca jest aktualizowana.

Uwaga

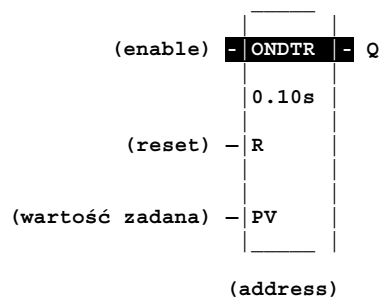
Jeżeli w programie sterującym znajdują się dwa lub więcej przełączników czasowych o tych samych przypisanych zmiennych, to ich bieżące wartości będą takie same.

Gdy bieżąca wartość równa się lub przekroczy wartość zadaną PV, na wyjściu Q zostaje przesłany sygnał. Przez cały czas dopływu sygnału do przełącznika, wartość ta jest inkrementowana, aż do momentu dojścia do wartości maksymalnej. Po dojściu do wartości maksymalnej 32767 zliczanie zostaje zatrzymane, a sygnał wyjściowy Q nie ulega zmianie.



- A = Na wejściu ENABLE pojawia się sygnał. Przełącznik rozpoczyna zliczanie
- B = Wartość bieżąca osiąga wartość zadaną PV. Na wyjściu Q wysyłany jest sygnał.
- C = Na wejście RESET doprowadzony zostaje sygnał. Na wyjściu Q nie jest przesyłany sygnał. Czas zliczany zostaje wyzerowany.
- D = Na wejście RESET nie jest przesyłany sygnał. Przełącznik rozpoczyna zliczanie czasu od początku.
- E = Sygnał przestaje być doprowadzany do wejścia Enable. Przełącznik przerywa zliczanie czasu. Wartość bieżąca pozostaje zapamiętana.
- F = Na wejście ENABLE doprowadzony zostaje ponownie sygnał. Przełącznik kontynuuje zliczanie czasu.
- G = Wartość bieżąca osiąga wartość zadaną PV. Na wyjściu wysyłany jest sygnał. Przełącznik kontynuuje zliczanie do momentu, kiedy na wejście ENABLE przestanie być doprowadzany sygnał, na wejście RESET zostanie doprowadzony sygnał lub wartość bieżąca CV będzie równa maksymalnemu czasowi.
- H = Na wejście ENABLE przestaje być podawany sygnał. Przełącznik przerywa zliczanie czasu.

W momencie gdy do przełącznika przestaje dopływać sygnał, zliczanie czasu zostaje zatrzymane i zapamiętany zostaje stan licznika. Jeżeli na wyjściu Q jest doprowadzany sygnał, jest on nadal doprowadzany. Jeżeli na wejście doprowadzony zostanie ponownie sygnał, wartość bieżąca jest ponownie inkrementowana, począwszy od zapamiętanej wcześniej wartości. Doprowadzenie sygnału do wejścia R powoduje ustawienie bieżącej wartości na 0 i zaprzestanie przesyłania sygnału na wyjściu Q. W sterownikach serii 35x i 36x, jeżeli do wejścia enable nie jest podawany sygnał, PV=0 i doprowadzany jest sygnał do wejścia R, na wyjściu nie jest przesyłany sygnał. Należy jednak pamiętać, że w przypadku jednostek centralnych 311-341, w takich samych warunkach, na wyjściu będzie przesyłany sygnał.



Parametry

Parametr	Opis
address	<p>Przełącznik czasowy ONDTR wykorzystuje trzy słowa (rejstry) pamięci typu %R, do zapamiętywania następujących parametrów:</p> <ul style="list-style-type: none"> • Wartość bieżąca (CV)= słowo 1 • Wartość zadana (PV)=słowo 2. • Słowo sterujące =słowo 3. <p>Dla przełącznika czasowego ONDTR, bezpośrednio pod blokiem funkcyjnym, należy podać adres początkowy obszaru pamięci o długości 3 słów (rejestrów).</p> <p>Uwaga: Adres ten nie powinien być wykorzystywany przez inne bloki funkcyjne.</p> <p>Ostrzeżenie: Wykorzystywanie tego adresu przez inne elementy logiczne powoduje zakłócenia w pracy przełącznika.</p>
enable	Sygnal wejściowy, uruchamiający funkcję zliczania czasu.
R	Sygnal zerujący wartość bieżącą.
PV	Wartość zadana, kopiowana z wejścia PV do drugiego rejestru (adres+1) w momencie zerowania lub uruchomienia przełącznika.
Q	Sygnal wyjściowy wysyłany jeżeli wartość bieżąca jest większa lub równa od wartości zadanej.
Czas	Czas zliczany w dziesiątych (0.1), setnych (0.01) lub tysięcznych (0.001) częściach sekundy.

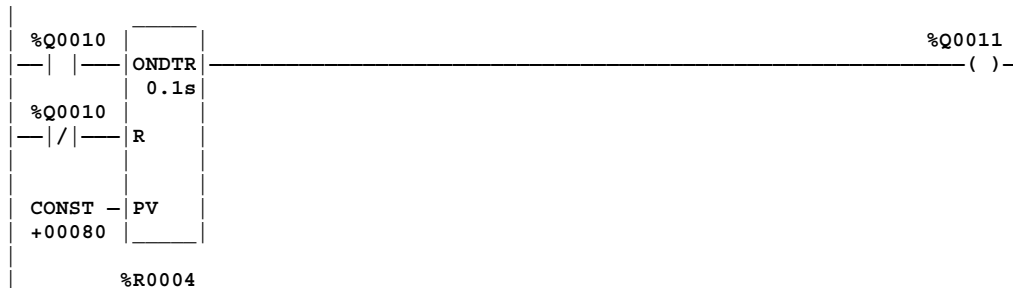
Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
adres								•				
enable	•											
R	•											
PV		•	•	•	•		•	•	•	•	•	•
Q	•											•

- Dozwolony typ parametru.

Przykład

W zamieszczonym poniżej przykładzie, przełącznik czasowy z pamięcią wykorzystany jest to wysłania sygnału (%Q0011) po upływie 8.0 sekund od momentu ustawienia zmiennej %Q0010 na wartość 1. Sygnał przestaje być wysyłany po ustawieniu zmiennej %Q0010 na 0.



Przełącznik czasowy załączający (TMR)

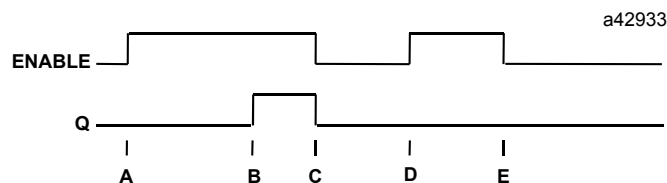
Przełącznik czasowy załączający (TMR) zlicza czas, gdy doprowadzany jest do niego sygnał, zostaje wyzerowany, gdy sygnał przestaje dopływać. Czas może być zliczany w dziesiątych, setnych lub tysięcznych częściach sekundy. Zakres dopuszczalnych wartości wynosi 0 do +32 767, a więc zliczany czas może wynosić od 0.001 do 3 276.7 s. Wartość bieżąca przełącznika jest przechowywana w przypadku awarii zasilania sterownika; nie jest przeprowadzana automatyczna inicjalizacja po włączeniu zasilania.

Po pierwszym doprowadzeniu sygnału do przełącznika, rozpoczyna on naliczanie czasu. Bieżąca wartość jest aktualizowana po każdym wywołaniu tego bloku funkcyjnego w programie sterującym, co pozwala na odczytanie czasu, który upłynął od momentu, kiedy na wejściu reset pojawił się sygnał.

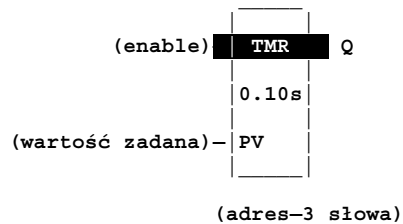
Uwaga

Jeżeli w programie sterującym znajdują się dwa lub więcej przełączników czasowych o tych samych przypisanych zmiennych, to ich bieżące wartości będą takie same.

Wartość jest zliczana tak długo, jak długo doprowadzany jest sygnał do tego przełącznika. Jeżeli wartość bieżąca jest równa lub większa od wartości zadanej PV, na wyjście Q wysyłany jest sygnał. Przełącznik kontynuuje odliczanie czasu, do momentu dojścia do wartości maksymalnej. Jeżeli do wejścia Enable przestanie dopływać sygnał, przełącznik wstrzymuje zliczanie czasu, a wartość bieżąca zostaje wyzerowana.



- A = Na wejście ENABLE doprowadzony zostaje sygnał. Przełącznik rozpoczyna zliczanie czasu.
- B = Wartość bieżąca osiąga wartość zadaną PV. Na wyjście Q wysyłany jest sygnał, a przełącznik nadal kontynuuje odliczanie czasu.
- C = Na wejście Enable przestaje być podawany sygnał, na wyjście nie jest przesyłany sygnał, zatrzymane zostaje zliczanie czasu oraz zerowana jest wartość bieżąca.
- D = Na wejście ENABLE ponownie doprowadzony zostaje sygnał. Przełącznik rozpoczyna zliczanie czasu.
- E = Na wejście ENABLE przestaje być podawany sygnał zanim wartość bieżąca osiągnie wartość zadaną. Na wyjście nie jest nadal przesyłany sygnał. Przełącznik przerywa zliczanie czasu, zerując wartość bieżącą.



Parametry

Parametr	Opis
adres	<p>Przełącznik czasowy TMR wykorzystuje trzy słowa (rejstry) pamięci typu %R, do zapamiętywania następujących parametrów:</p> <ul style="list-style-type: none"> • Wartość bieżąca (CV) = słowo 1. • Wartość zadanej (PV) = słowo 2. • Słowo sterujące = słowo 3. <p>Dla przełącznika czasowego TMR, bezpośrednio pod blokiem funkcyjnym, należy podać adres początkowy obszaru pamięci o długości 3 słów (rejestrów).</p> <p>Uwaga: Adres ten nie powinien być wykorzystywany przez inne bloki funkcyjne.</p> <p>Ostrzeżenie: Wykorzystywanie tego adresu przez inne elementy logiczne powoduje zakłócenia w pracy przełącznika.</p>
enable	Sygnal wejściowy, uruchamiający funkcję zliczania czasu. Jeżeli do wejścia Enable nie jest doprowadzany sygnał, wartość bieżąca jest zerowana a na wyjście Q nie jest przesyłany sygnał.
PV	Wartość zadana, kopiowana z wejścia PV do drugiego rejestru (adres+1) w momencie zerowania lub uruchomienia przełącznika.
Q	Sygnal wyjściowy wysyłany jeżeli do wejścia przełącznika TMR dopływa sygnał, a wartość bieżąca jest większa lub równa od wartości zadanej.

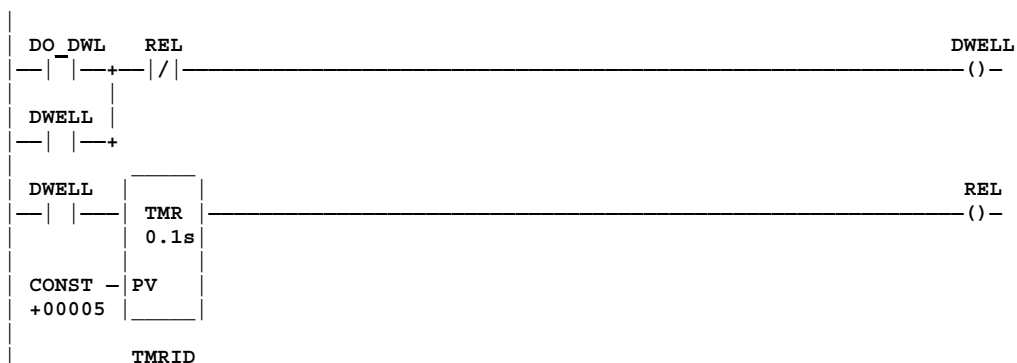
Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
adres								•				
enable	•											
PV		•	•	•	•		•	•	•	•	•	•
Q	•											•

- Dozwolony typ parametru.

Przykład

W zamieszczonym poniżej przykładzie, przekaźnik czasowy (z adresem) TMRID wykorzystywany jest do sterowania czasem załączenia przekaźnika DWELL. Do przekaźnika DWELL doprowadzany jest sygnał, jeżeli zmienna DO_DWL przypisana do styku otwartego ma wartość 1. Styk przekaźnika DWELL podtrzymuje sygnał przekaźnika DWELL (po rozwarciu styku DO-DWELL) oraz powoduje również uruchomienie przekaźnika czasowego TMRID. Jeżeli wartość bieżąca przekaźnika TMRID będzie równa pół sekundy, doprowadzony zostanie sygnał do przekaźnika REL, co spowoduje przerwanie dopływu sygnału do przekaźnika DWELL. Styk DWELL przerywa dopływ sygnału do TMRID, wyzerowanie jego bieżącej wartości oraz przerwanie dopływu sygnału do przekaźnika REL. Obwód jest w tym momencie gotowy do następnego, zwarcia styku DO_DWL.



Przełącznik czasowy wyłączający (OFDT)

Przełącznik czasowy wyłączający (OFDT) zlicza czas, gdy nie dopływa do niego sygnał i zostaje wyzerowany, gdy sygnał zacznie dopływać. Czas może być zliczany w dziesiątych, setnych lub tysięcznych częściach sekundy. Zakres zmierzonej wartości wynosi od 0 do +32767 jednostek czasu. Wartość bieżąca przełącznika jest przechowywana w przypadku awarii zasilania sterownika; nie jest przeprowadzana automatyczna inicjalizacja po włączeniu zasilania.

Po pierwszym doprowadzeniu sygnału do przełącznika, wartość bieżąca zostaje ustawiona na zero i sygnał zostaje przesłany na wyjście Q. (Przełącznik czasowy OFDT wykorzystuje słowo 1 (rejestr) do przechowywania wartości bieżącej CV - proszę porównać z punktem "Blok danych sterujących"). Sygnał podawany jest na wyjście tak długo, jak długo doprowadzany jest sygnał do tego bloku funkcyjnego. Gdy sygnał przestanie dopływać, na wyjście nadal przekazywany jest sygnał oraz następuje rozpoczęcie zliczania czasu.

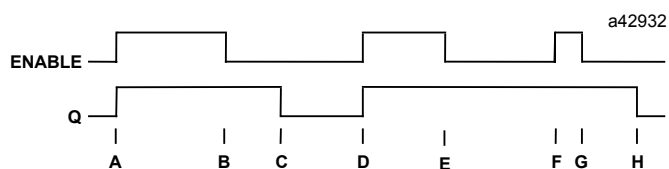
Uwaga

Jeżeli w programie sterującym znajdują się dwa lub więcej przełączników czasowych o tych samych przypisanych zmiennych, to ich bieżące wartości będą takie same.

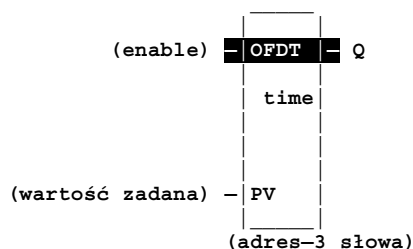
Przełącznik nie przesyła sygnału na wyjście, w przypadku gdy parametr PV jest równy zero lub ma wartość ujemną.

Jeżeli wykonany zostanie ten blok funkcyjny, a do wejścia enable nie jest doprowadzany sygnał, wartość bieżąca zostaje zaktualizowana i będzie ona podawać czas, który upłynął od momentu zaprzestania doprowadzania sygnału do przełącznika. Jeżeli wartość bieżąca CV jest równa lub większa od wartości zadanej PV, zostaje wstrzymane przesyłanie sygnału na wyjście Q. Przełącznik zatrzymuje wtedy odliczanie czasu - proszę porównać z Częścią C wykresu zamieszczonego poniżej.

Przy ponownym doprowadzeniu sygnału, wartość bieżąca zostaje wyzerowana.



- A = Na wejściu ENABLE pojawia się sygnał. Przełącznik zostaje wyzerowany (CV=0), a na wyjściu również pojawia się sygnał.
- B = Na wejście ENABLE przestaje być podawany sygnał. Przełącznik rozpoczyna zliczanie czasu.
- C = Wartość bieżąca CV osiąga wartość zadaną PV. Na wyjście nie jest przesyłany sygnał. Przełącznik przerywa zliczanie czasu.
- D = Na wejściu enable pojawia się sygnał. Przełącznik zostaje wyzerowany (CV=0).
- E = Na wejście ENABLE przestaje być podawany sygnał. Przełącznik rozpoczyna zliczanie czasu.
- F = Na wejściu ENABLE pojawia się sygnał. Przełącznik zostaje wyzerowany (CV=0).
- G = Na wejście ENABLE przestaje być podawany sygnał. Przełącznik rozpoczyna zliczanie czasu.
- H = Wartość bieżąca CV osiąga wartość zadaną PV. Na wyjście nie jest przesyłany sygnał. Przełącznik przerywa zliczanie czasu.



Gdy przekaźnik czasowy OFDT jest wykorzystywany w bloku programu, który *nie jest* wywoływany w każdym cyklu pracy sterownika, zlicza on czas pomiędzy kolejnymi wywołaniami bloku programu, aż do momentu wyzerowania. Oznacza to, że pracuje on w programie jako przekaźnik o znacznie dłuższym cyklu niż przekaźnik w głównym bloku programu. Jeżeli więc blok programu będzie nieaktywny przez dłuższy czas, przekaźnik powinien być tak zaprogramowany, aby skorzystać z tej jego cechy. Przykładowo, jeżeli przekaźnik w bloku programu jest wyzerowany i ten blok programu nie jest wywoływany (jest nieaktywny) przez cztery minuty, po wywołaniu bloku, odliczone zostaną cztery minuty. Jest to czas, który upłynął od momentu zaprzestania doprowadzenia sygnału do przekaźnika, o ile nie został on wcześniej wyzerowany.

Parametry

Parametr	Opis
adres	<p>Przekaźnik czasowy OFDT wykorzystuje trzy słowa (rejstry) pamięci typu %R, do zapamiętywania następujących parametrów:</p> <ul style="list-style-type: none"> • Wartość bieżąca (CV) = słowo 1. • Wartość zadana (PV) = słowo 2. • Słowo sterujące = słowo 3. <p>Dla przekaźnika czasowego OFDT, bezpośrednio pod blokiem funkcyjnym, należy podać adres początkowy obszaru pamięci o długości 3 słów (rejestrów).</p> <p>Uwaga: Adres ten nie powinien być wykorzystywany przez inne bloki funkcyjne.</p> <p>Ostrzeżenie: Wykorzystywanie tego adresu przez inne elementy logiczne powoduje zakłócenia w pracy przekaźnika.</p>
enable	Sygnał wejściowy, uruchamiający funkcję zliczania czasu.
Czas	Czas zliczany w dziesiątych (0.1), setnych (0.01) lub tysięcznych (0.001) częściach sekundy.
PV	Wartość zadana, kopiowana z wejścia PV do drugiego rejestru (adres+1) w momencie zerowania lub uruchomienia przekaźnika.
Q	Sygnał wyjściowy wysyłany jeżeli wartość bieżąca jest mniejsza od wartości ustawionej. Sygnał ten jest przechowywany w przypadku awarii zasilania sterownika; nie jest przeprowadzana automatyczna inicjalizacja po włączeniu zasilania.

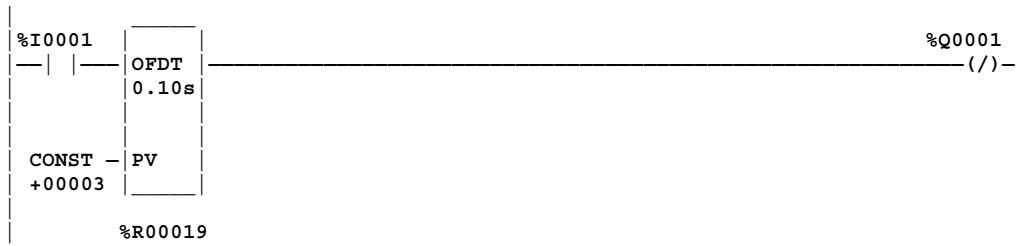
Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
adres								•				
enable	•											
PV	•	•	•	•	•		•	•	•	•	•	•
Q	•											•

- Dozwolony typ parametru.

Przykład

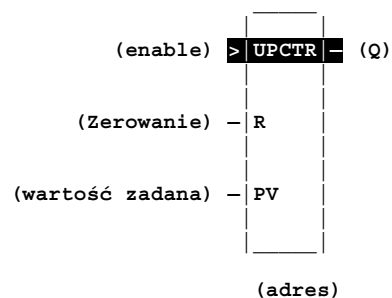
W zamieszczonym poniżej przykładzie, przełącznik czasowy OFDT zeruje parametr wyjściowy (%Q0001) zawsze, ilekroć wartość zmiennej związanej z wejściem (%I0001) zmieni się na 1. Sygnał wyjściowy jest ponownie doprowadzany po upływie 0.3 sekundy od momentu ustawienia wejścia na 0.



Licznik zliczający w górę (UPCTR)

Licznik zliczający w górę służy do zliczania impulsów sygnału od 0 do zadanej wartości. Zakres licznika wynosi od 0 do +32767 impulsów. Zbocze narastające sygnału wejściowego (zmiana stanu sygnału wejściowego z 0 na 1) powoduje zwiększenie wartości bieżącej o 1. Wartość ta może być zwiększana ponad wartość zadaną PV. Sygnał wyjściowy jest wysyłany zawsze, gdy wartość bieżąca jest większa lub równa od wartości zadanej.

Wartość bieżąca licznika jest przechowywana w przypadku awarii zasilania sterownika, nie jest przeprowadzana automatyczna inicjalizacja po włączeniu zasilania.



Parametry

Parametr	Opis
adres	<p>Licznik zliczający w górę UPCTR wykorzystuje trzy słowa (rejstry) pamięci typu %R, do zapamiętywania następujących parametrów:</p> <ul style="list-style-type: none"> • Wartość bieżąca (CV) = słowo 1. • Wartość zadana (PV) = słowo 2. • Słowo sterujące = słowo 3. <p>Dla licznika UPCTR, bezpośrednio pod blokiem funkcyjnym, należy podać adres początkowy obszaru pamięci o długości 3 słów (rejestrów).</p> <p>Uwaga: Adres ten nie powinien być wykorzystywany przez inne bloki funkcyjne - w przeciwnym razie blok funkcyjny będzie nieprawidłowo wykonywany.</p> <p>Ostrzeżenie: Wykorzystywanie tego adresu przez inne elementy logiczne powoduje zakłócenia w pracy licznika.</p>
enable	Sygnał wejściowy. Każde zbocze narastające sygnału wejściowego powoduje wzrost wartości bieżącej licznika o 1.
R	Sygnał zerujący wartość bieżącą.
PV	Wartość zadana, kopiowana z wejścia PV do drugiego rejestru (adres+1) w momencie zerowania lub uruchomienia licznika.
Q	Sygnał wyjściowy jest wysyłany jeżeli wartość bieżąca jest większa lub równa od wartości zadanej.

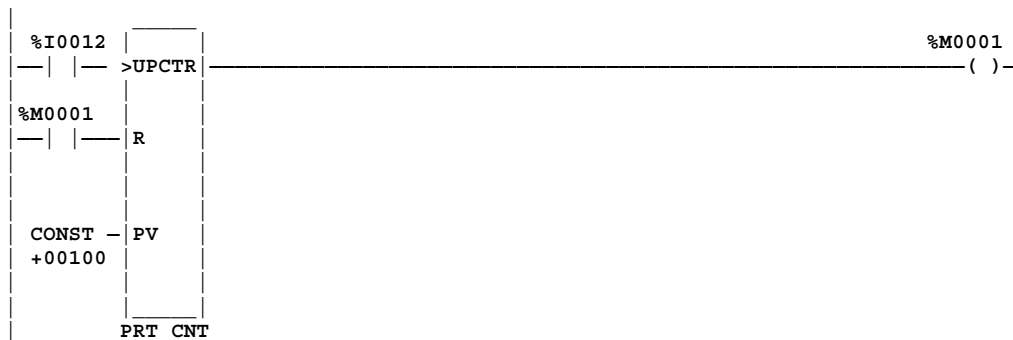
Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
adres								•				
enable	•											
R	•											
PV		•	•	•	•		•	•	•	•	•	•
Q	•											•

- Dozwolony typ parametru.

Przykład

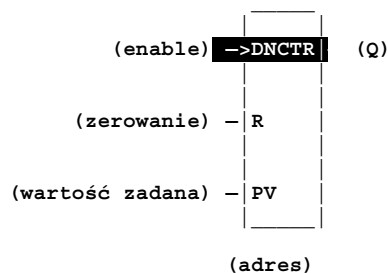
W zamieszczonym poniżej przykładzie, każde zbocze narastające sygnału wejściowego %I0012 powoduje zwiększenie wartości licznika PRT_CNT o 1. Sygnal jest doprowadzany do wewnętrznego przekaźnika %M001 zawsze po odliczeniu 100 impulsów. Ustawienie wartości zmiennej %M001 na 1 powoduje wyzerowanie wartości zliczanej przez licznik.



Licznik zliczający w dół (DNCTR)

Licznik zliczający w dół (DNCTR) służy do odliczania impulsów sygnału od zadanej wartości do 0. Minimalna wartość zadana może być równa zero, a maksymalna +32 767 impulsów. Minimalna wartość bieżąca wynosi -32 768. Podanie sygnału na wejście zerujące powoduje skopiowanie wartości bieżącej do rejestru, w którym przechowywana jest wartość zadana. Zbocze narastające sygnału wejściowego (zmiana stanu sygnału wejściowego z 0 na 1) powoduje dekrementowanie wartości bieżącej o 1. Sygnal wyjściowy jest wysyłany, gdy wartość bieżąca jest większa lub równa zero.

Wartość bieżąca licznika jest przechowywana w przypadku awarii zasilania sterownika, nie jest przeprowadzana automatyczna inicjalizacja po włączeniu zasilania.



Parametry

Parametr	Opis
adres	<p>Licznik zliczący w dół DNCTR wykorzystuje trzy słowa (rejstry) pamięci typu %R, do zapamiętywania następujących parametrów:</p> <ul style="list-style-type: none"> • Wartość bieżąca (CV) = słowo 1. • Wartość zadana (PV) = słowo 2 • Słowo sterujące = słowo 3. <p>Dla licznika DNCTR, bezpośrednio pod blokiem funkcyjnym, należy podać adres początkowy obszaru pamięci o długości 3 słów (rejestrów).</p> <p>Uwaga: Adres ten nie powinien być wykorzystywany przez inne bloki funkcyjne - w przeciwnym razie blok funkcyjny będzie nieprawidłowo wykonywany.</p> <p>Ostrzeżenie: Wykorzystywanie tego adresu przez inne elementy logiczne powoduje zakłócenia w pracy licznika.</p>
enable	Sygnał wejściowy. Każde zbocze narastające sygnału wejściowego powoduje dekrementowanie wartości bieżącej licznika o 1.
R	Sygnał powodujący ustawienie wartości bieżącej na wartość zadaną.
PV	Wartość zadana, kopiowana z wejścia PV do drugiego rejestru (adres+1) w momencie zerowania lub uruchomienia licznika.
Q	Sygnał wyjściowy wysyłany jeżeli wartość bieżąca jest większa lub równa zero.

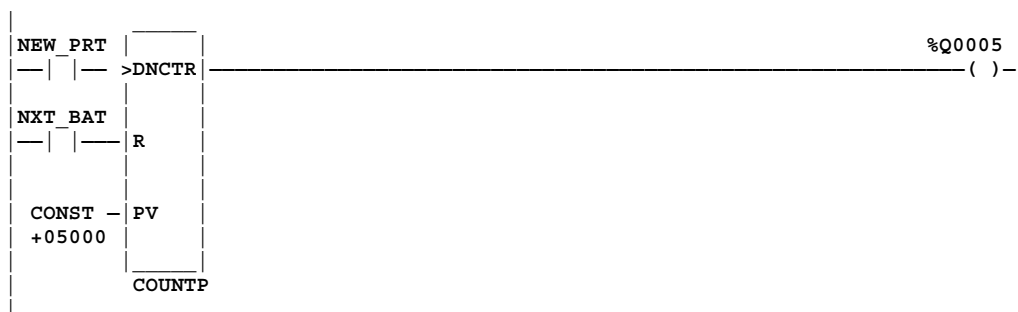
Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
adres								•				
enable	•											
R	•											
PV		•	•	•	•		•	•	•	•	•	•
Q	•											•

- Dozwolony typ parametru.

Przykład

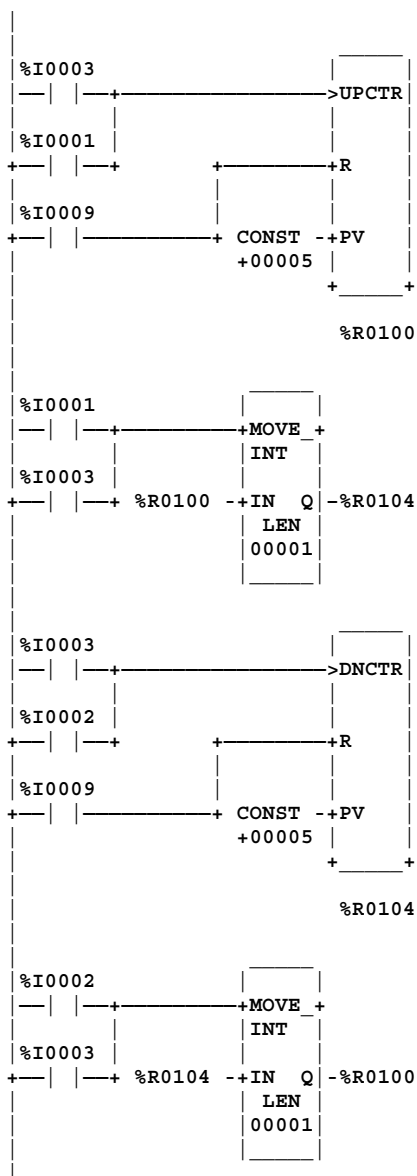
W zamieszczonym poniżej przykładzie, licznik odejmujący COUNTP odlicza 5000 nowych impulsów przed ustawieniem wartości zmiennej %Q0005 na 1.



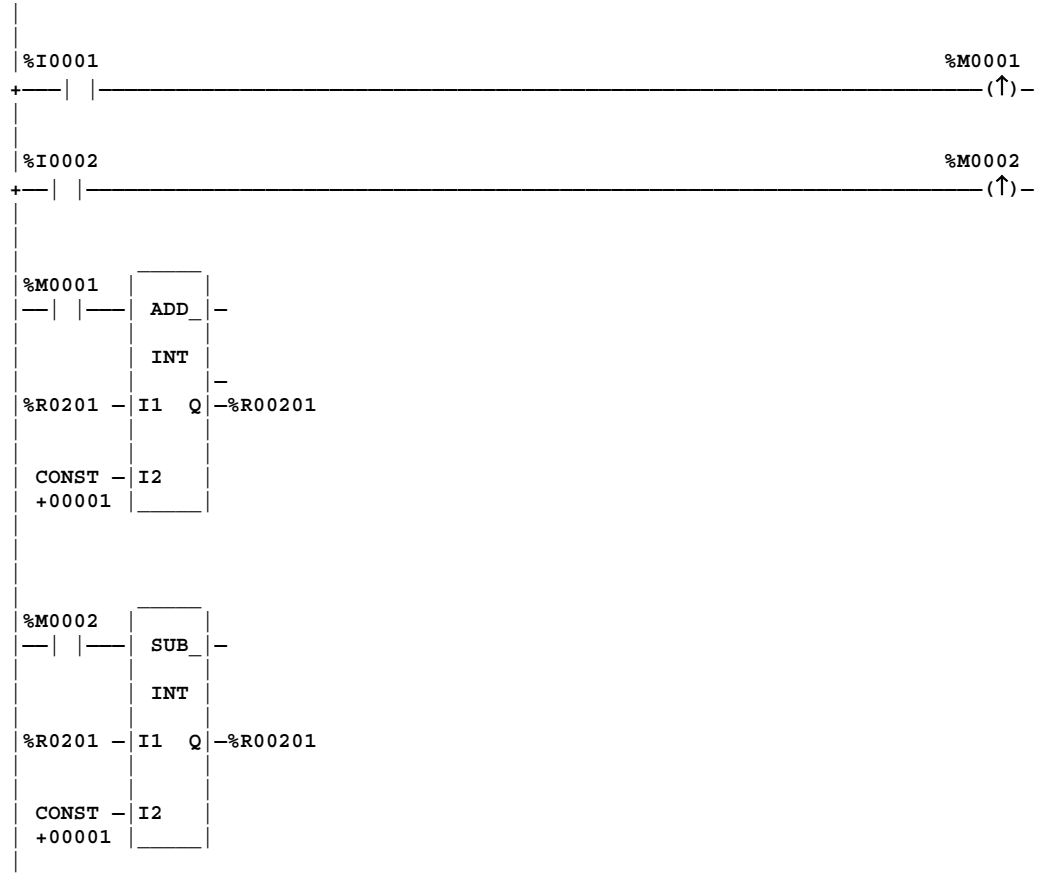
Przykład

W zamieszczonym poniżej przykładzie sterownik zlicza liczbę przedmiotów zgromadzonych w magazynie przejściowym. Istnieją dwa sposoby rozwiązania tego zadania za pomocą zestawu instrukcji sterowników 90-30/20/Micro.

Pierwsza metoda polega na zastosowaniu pary liczników, dodającego i odejmującego, korzystających z jednego rejestru do przechowywania wartości bieżącej. Po przetransportowaniu przedmiotu do magazynu, wartość licznika dodającego jest zwiększana o 1, powodując tym samym zwiększenie liczby przedmiotów przechowywanych w magazynie o 1. Jeżeli natomiast przedmiot jest wydawany z magazynu, wartość bieżąca licznika odejmującego jest dekrementowana o 1, w celu odzwierciedlenia aktualnego stanu magazynu. W przypadku zmiany wartości bieżącej CV licznika, jego aktualna wartość musi być kopiowana do rejestru z wartością bieżącą drugiego licznika.



Drugie rozwiązanie tego zadania polega na zastosowaniu funkcji ADD i SUB.



W rozdziale tym opisano funkcje matematyczne, dostępne w zestawie instrukcji sterowników 90-30/20/Micro.

Oznaczenie skrótowe	Funkcja	Opis	Strona
ADD	Dodawanie	Dodawanie dwóch liczb.	6-1
SUB	Odejmowanie	Odejmowanie dwóch liczb	6-1
MUL	Mnożenie	Mnożenie dwóch liczb.	6-1
DIV	Dzielenie bez reszty	Część całkowita z dzielenia dwóch liczb.	6-1
MOD	Dzielenie modulo	Reszta z dzielenia dwóch liczb.	6-6
SQRT	Pierwiastek kwadratowy	Obliczanie pierwiastka kwadratowego z liczby całkowitej lub rzeczywistej.	6-8
SIN, COS, TAN, ASIN, ACOS, ATAN	Funkcje trygonometryczne†	Obliczenie odpowiedniej funkcji dla parametru będącego liczbą rzeczywistą, zadanego parametrem wejściowym IN.	6-10
LOG, LN EXP, EXPT	Funkcje logarytmiczne/ wykładnicze †	Obliczenie odpowiedniej funkcji dla parametru będącego liczbą rzeczywistą, zadanego parametrem wejściowym IN.	6-12
RAD, DEG	Konwersja wartości kąta	Obliczenie odpowiedniej funkcji dla parametru będącego liczbą rzeczywistą, zadanego parametrem wejściowym IN.	6-14

† Funkcje trygonometryczne, logarytmiczne/ wykładnicze oraz funkcje do konwersji wartości kąta mogą być realizowane **wyłącznie** w jednostkach centralnych 35x i 36x, oprogramowanie systemowe wer. 9 lub nowsza oraz we wszystkich wersjach jednostki centralnej CPU352.

Uwaga

Funkcje DIV i MOD są bardzo podobne, różnica pomiędzy nimi polega na zwracanych wartościach. Funkcja DIV zwraca część całkowitą wyniku dzielenia, a funkcja MOD resztę z dzielenia.

Standardowe funkcje matematyczne (ADD, SUB, MUL, DIV)

Funkcje te wykonują cztery podstawowe działania matematyczne: dodawanie, odejmowanie, mnożenie i dzielenie. Po doprowadzeniu sygnału do funkcji, wykonywane jest odpowiednie działanie matematyczne na dwóch liczbach I1 i I2, które są parametrami wejściowymi. Obydwa parametry wejściowe muszą być takiego samego typu. Parametr wyjściowy Q jest też tego samego typu.

Uwaga

Funkcja DIV nie zaokrągla wyniku do najbliższej liczby całkowitej.
(Przykładowo, $24 \text{ DIV } 5 = 4$.)

Funkcje matematyczne wykonują operacje na następujących typach danych:

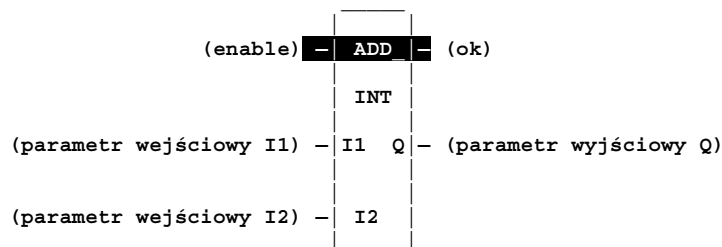
Typ danych	Opis
INT	Liczba całkowita ze znakiem (16 bitowa)
DINT	Liczba całkowita podwójnej precyzji ze znakiem (32 bitowa).
REAL	Liczba rzeczywista

Uwaga

Operacje na liczbach rzeczywistych można realizować wyłącznie w jednostkach centralnych 35x i 36x, oprogramowanie systemowe ver. 9 lub nowsza oraz we wszystkich wersjach jednostki centralnej CPU352.

Domyślnym typem danych (wybieranym automatycznie przez oprogramowanie) jest INT, jednakże może on zostać zmieniony po dokonaniu wyboru bloku funkcyjnego i wprowadzeniu go do szczebla drabiny logicznej programu sterującego. Więcej informacji o typach danych podano w Rozdziale 2 "Organizacja programu. Typy danych i typy zmiennych."

Jeśli wynik operacji przekracza dopuszczalny zakres wartości dla stosowanego typu danych, parametr wyjściowy przyjmuje największą dopuszczalną wartość, a sygnał wyjściowy nie jest przesyłany. Dla liczb ze znakiem, znak jest ustawiany tak, aby wskazywał kierunek przekroczenia zakresu. Jeśli wynik operacji mieści się w dopuszczalnym zakresie wartości, blok funkcyjny przesyła sygnał wyjściowy. W przypadku gdy parametry wejściowe są liczbami całkowitymi ze znakiem lub liczbami całkowitymi podwójnej precyzji, znak wyniku operacji DIV i MUL zależy od znaków parametrów wejściowych I1 i I2.



Parametry

Parametr	Opis
enable	Sygnal wejściowy. Doprowadzenie tego sygnału powoduje wykonanie funkcji.
I1	Wartość stała lub adres zmiennej, będącej pierwszym parametrem wykonywanego działania matematycznego. (Parametr I1 znajduje się z lewej strony działania matematycznego, jak na przykład w I1- I2).
I2	Wartość stała lub adres zmiennej, będącej drugim parametrem wykonywanego działania matematycznego. (Parametr I2 znajduje się z prawej strony działania matematycznego, jak na przykład w I1- I2).
ok	Sygnal wyjściowy wysyłany, jeżeli działanie zostało poprawnie wykonane, jego wynik mieści się w dopuszczalnym przedziale wartości i nie ma próby wykonania działania niewykonalnego.
Q	Wynik działania.

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
I1		o	o	o	o		o	•	•	•	•†	
I2		o	o	o	o		o	•	•	•	•†	
ok	•											•
Q		o	o	o	o		o	•	•	•		

- Typ może być wykorzystany jako dany parametr bloku funkcyjnego.
- o Typ może być wykorzystany wyłącznie jako parametr do operacji na danych typu INT. Nie można go wykorzystywać w przypadku operacji na danych typu DINT i REAL.
- † W operacjach wykonywanych na danych typu DINT, wartości stałe są ograniczone do przedziału <-32768, +32767>.

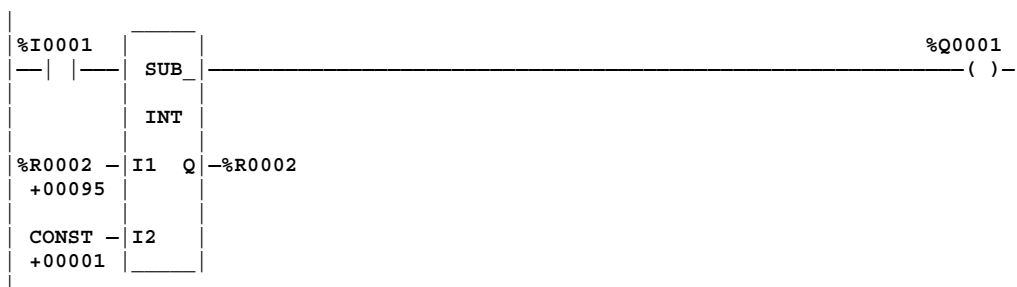
Uwaga

Domyślnym typem danych dla funkcji operujących na liczbach 16 bitowych lub pojedynczych rejestrach jest typ INT. W celu zmiany tego typu na typ DINT, podwójne słowo o długości 32 bitów lub REAL (możliwość dostępna wyłącznie w jednostkach centralnych 35x i 36x), należy wcisnąć **F10**. Wartości typu INT zajmują pojedynczy, 16 bitowy rejestr, %R, %AI lub %AQ.

Wartości typu DINT zajmują dwa sąsiadujące ze sobą rejestry, w pierwszym z nich przechowywane jest 16 młodszych bitów, a w drugim 16 starszych bitów. W jednostkach centralnych serii 35x i 36x (oprogramowanie systemowe wer. 9 lub nowsza), wartości typu REAL zajmują również dwa rejestry, w sumie 32 bity, znak przechowywany jest w najstarszym bicie, a następnie pamiętane są wykładnik i podstawa.

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0001 na 1 powoduje zmniejszenie wartości zmiennej %R0002 o 1 oraz ustawienie zmiennej przełącznika %Q0001 na 1, pod warunkiem, że w czasie wykonywania operacji odejmowania nie przekroczono dopuszczalnego zakresu wartości.



Funkcje matematyczne a typy danych

Element	Działanie	Format wyświetlania
ADD INT	$Q(16 \text{ bit}) = I1(16 \text{ bitów}) + I2(16 \text{ bitów})$	Liczba 10 cyfrowa ze znakiem o podstawie 5 cyfrowej
ADD DINT	$Q(32 \text{ bit}) = I1(32 \text{ bitów}) + I2(32 \text{ bitów})$	Liczba 10 cyfrowa ze znakiem o podstawie 8 cyfrowej
ADD REAL*	$Q(32 \text{ bit}) = I1(32 \text{ bitów}) + I2(32 \text{ bitów})$	Liczba 10 cyfrowa ze znakiem i częścią dziesiętną o podstawie 7 cyfrowej
SUB INT	$Q(16 \text{ bit}) = I1(16 \text{ bitów}) - I2(16 \text{ bitów})$	Liczba 10 cyfrowa ze znakiem o podstawie 5 cyfrowej
SUB DINT	$Q(32 \text{ bit}) = I1(32 \text{ bitów}) - I2(32 \text{ bitów})$	Liczba 10 cyfrowa ze znakiem o podstawie 8 cyfrowej
SUB REAL*	$Q(32 \text{ bit}) = I1(32 \text{ bitów}) - I2(32 \text{ bitów})$	Liczba 10 cyfrowa ze znakiem i częścią dziesiętną o podstawie 7 cyfrowej
MUL INT	$Q(16 \text{ bitów}) = I1(16 \text{ bitów}) * I2(16 \text{ bitów})$	Liczba 10 cyfrowa ze znakiem o podstawie 5 cyfrowej
MUL DINT	$Q(32 \text{ bitów}) = I1(32 \text{ bitów}) * I2(32 \text{ bitów})$	Liczba 10 cyfrowa ze znakiem o podstawie 8 cyfrowej
MUL REAL*	$Q(32 \text{ bitów}) = I1(32 \text{ bitów}) * I2(32 \text{ bitów})$	Liczba 10 cyfrowa ze znakiem i częścią dziesiętną o podstawie 7 cyfrowej
DIV INT	$Q(16 \text{ bitów}) = I1(16 \text{ bitów}) / I2(16 \text{ bitów})$	Liczba 10 cyfrowa ze znakiem o podstawie 5 cyfrowej
DIV DINT	$Q(32 \text{ bitów}) = I1(32 \text{ bitów}) / I2(32 \text{ bitów})$	Liczba 10 cyfrowa ze znakiem o podstawie 8 cyfrowej
DIV REAL*	$Q(32 \text{ bitów}) = I1(32 \text{ bitów}) / I2(32 \text{ bitów})$	Liczba 10 cyfrowa ze znakiem i częścią dziesiętną o podstawie 7 cyfrowej

* dotyczy wyłącznie jednostek centralnych 35x i 36x, oprogramowanie systemowe wer. 9 lub nowsza oraz wszystkich wersji jednostki centralnej CPU352

Uwaga

Typy danych wejściowych i wyjściowych nie muszą być takie same. Funkcje MUL i DIV nie pozwalają na mieszanie typów danych, jak to ma miejsce w przypadku sterowników 90-70. Przykładowo, wykonanie działania MUL INT na dwóch liczbach 16 bitowych daje w wyniku liczbę 16 bitową, a nie 32 bitową. Aby w wyniku działania funkcji MUL DINT otrzymać liczbę 32 bitową, obydwa parametry wejściowe muszą być liczbami 32 bitowymi. W wyniku działania funkcji DIV INT, której parametrami są liczby 16 bitowe otrzymywana jest liczba 16 bitowa, natomiast w wyniku działania funkcji DIV DINT, której parametrami są liczby 32 bitowe otrzymywana jest liczba 32 bitowa.

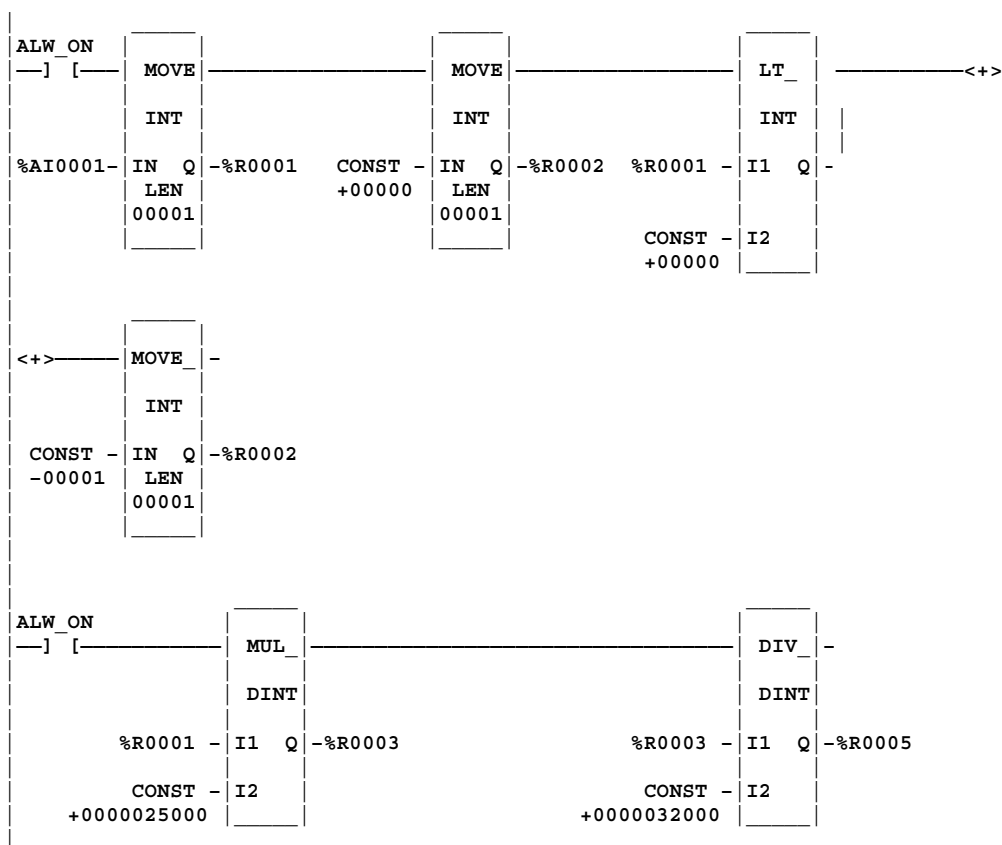
Funkcje te przesyłają sygnał wyjściowy, pod warunkiem, że nie nastąpi przekroczenie dopuszczalnego zakresu wartości. W przypadku przekroczenia dopuszczalnego zakresu wartości, wynik jest równy najwyższej, dopuszczalnej wartości i posiada poprawny znak, ale nie jest wysyłany sygnał wyjściowy.

Należy zwrócić uwagę, aby przy stosowaniu funkcji MUL i DIV nie nastąpiło przekroczenie dopuszczalnego zakresu wartości. Przy konwertowaniu typu INT na DINT należy pamiętać, że jednostka centralna korzysta ze standardowego formatu liczby: dopełnienia do dwóch ze znakiem. Należy sprawdzić znak młodszego słowa 16 bitowego i zastosować go do drugiego słowa 16 bitowego. Jeżeli najbardziej znaczący bit w 16 bitowym słowie INT jest równy 0 (dodatni), należy przesunąć 0 do drugiego słowa. Jeżeli najbardziej znaczący bit w 16 bitowym słowie INT jest równy -1 (ujemny), należy przesunąć 1 do drugiego słowa. Konwersja typu DINT na INT jest łatwiejsza, ponieważ młodsze, 16 bitowe słowo jest częścią INT 32 bitowego słowa DINT. Starsze 16 bitów lub drugie słowo, powinny być równe 0 (wartość dodatnia) lub -1 (wartość ujemna), a jeżeli tak nie jest, liczba jest zbyt duża aby można ją było przekonwertować na liczbę 16 bitową.

Przykład

Powszechnie spotkanym zastosowaniem jest skalowanie wartości wejścia analogowego przy pomocy działania MUL i następnie działania DIV oraz ewentualnie ADD. Przy zakresie do 32000, zastosowanie funkcji MUL INT spowoduje przekroczenie dopuszczalnego zakresu wartości. Zastosowanie w funkcji MUL DINT zmiennej typu %AI również nie jest możliwe, ponieważ 32 bitowy parametr wejściowy I1 spowoduje jednoczesne połączenie 2 wejść analogowych. Wartość wejścia analogowego należy przesunąć do młodszego słowa rejestru 32 bitowego, sprawdzić znak, a następnie ustawić wartość drugiego rejestru na 0 jeżeli jest to liczba dodatnia lub na -1 jeżeli jest to liczba ujemna. Jeżeli do funkcji DIV ma być doprowadzony 32 bitowy parametr wejściowy, poprzedzającą funkcja MUL DINT musi mieć podane parametry w podwójnych rejestrach.

Przykładowo, zamieszczony poniżej program sterujący pozwala na przeskalowanie wartości wejścia analogowego %AI1 (-10...+10 V) na jednostki inżynierskie przechowywane w zmiennej %R5 (-25000...+25000 jednostek inżynierskich).



Dzielenie modulo (MOD) (INT, DINT)

Funkcja Modulo (MOD) dzieli dwie liczby i zwraca resztę z dzielenia. Znak wyniku jest zawsze taki sam jak znak parametru wejściowego I1.

Parametry funkcji MOD muszą mieć taki sam typ:

Typ	Opis
INT	Liczba całkowita ze znakiem (16 bitowa)
DINT	Liczba całkowita podwójnej precyzji ze znakiem (32 bitowa).

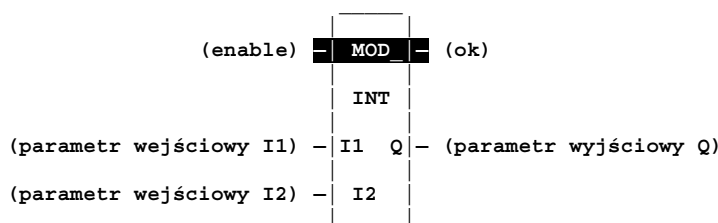
Domyślnym typem danych (wybieranym automatycznie przez oprogramowanie) jest INT, jednakże może on zostać zmieniony po dokonaniu wyboru bloku funkcyjnego i wprowadzeniu go do szczebla drabiny logicznej programu sterującego. Więcej informacji o typach danych podano w Rozdziale 2 w punkcie "Organizacja programu. Typy danych i typy zmiennych."

Doprowadzenie sygnału do bloku funkcyjnego powoduje podzielenie parametru wejściowego I1 przez I2. Obydwa parametry wejściowe muszą być takiego samego typu. Wynik (parametr Q) jest obliczany według podanego poniżej wzoru:

$$Q = I1 - ((I1 \text{ DIV } I2) * I2)$$

gdzie wynikiem dzielenia DIV jest liczba całkowita. Parametr wyjściowy Q reprezentuje ten sam typ danych, co parametry wejściowe I1 oraz I2.

Sygnał wyjściowy jest przesyłany zawsze po otrzymaniu sygnału wejściowego, chyba że nastąpi próba dzielenia przez zero. W takim wypadku sygnał wyjściowy nie jest przesyłany.



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie funkcji.
I1	I1 - wartość stała lub adres zmiennej, będącej dzielnią.
I2	I2 - wartość stała lub adres zmiennej, będącej dzielnikiem.
ok	Sygnał wyjściowy wysyłany, jeżeli działanie zostało poprawnie wykonane, a jego wynik mieści się w dopuszczalnym przedziale wartości.
Q	Wynik działania - reszta z dzielenia I1 przez I2.

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
I1		o	o	o	o		o	•	•	•	•†	
I2		o	o	o	o		o	•	•	•	•†	
ok	•											•
Q		o	o	o	o		o	•	•	•		

- Typ może być wykorzystany jako dany parametr bloku funkcyjnego.
- o Dopuszczalne jest stosowanie wyłącznie zmiennych typu INT, nie można stosować zmiennych typu DINT.
- † W przypadku operacji wykonywanych na zmiennych typu DINT, wartości stałe są ograniczone do przedziału <-32768, +32767>.

Przykład

W zamieszczonym poniżej przykładzie, zawsze, gdy zmienna %I0001 jest równa 1, reszta z dzielenia zmiennej BOXES przez Pallets zapisywana jest do zmiennej NT_FULL.

```

%I0001  MOD_
---| |---
      INT
PALLETs- I1 Q - NT_FULL
-00017   -0005
      BOXES - I2
+00006   _____

```

Pierwiastek kwadratowy (SQRT) (INT, DINT, REAL)

Blok funkcyjny SQRT wylicza pierwiastek kwadratowy z liczby. Gdy do bloku dociera sygnał, parametr wyjściowy Q przyjmuje wartość równą części całkowitej pierwiastka z liczby zadanej parametrem wejściowym IN. Parametr Q musi reprezentować taki sam typ danych jak parametr IN.

Funkcja SQRT operuje na następujących typach danych:

Typ	Opis
INT	Liczba całkowita ze znakiem (16 bitowa)
DINT	Liczba całkowita podwójnej precyzji ze znakiem (32 bitowa).
REAL	Liczba rzeczywista

Uwaga

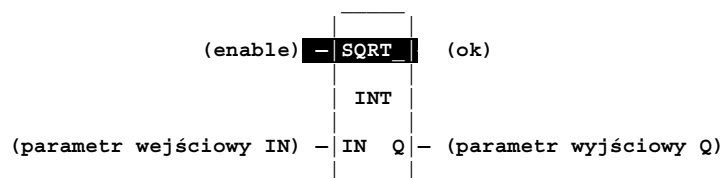
Operacje na liczbach rzeczywistych można realizować wyłącznie w jednostkach centralnych 35x i 36x, oprogramowanie systemowe ver. 9 lub nowsza oraz we wszystkich wersjach jednostki centralnej CPU352.

Domyślnym typem danych (wybieranym automatycznie przez oprogramowanie) jest INT, jednakże może on zostać zmieniony po dokonaniu wyboru bloku funkcyjnego i wprowadzeniu go do szczebla drabiny logicznej programu sterującego. Więcej informacji o typach danych podano w Rozdziale 2 w punkcie "Organizacja programu. Typy danych i typy zmiennych."

Sygnał wyjściowy jest przesyłany, gdy operacja zostanie wykonana bez przekroczenia dopuszczalnego zakresu wartości oraz nie miała miejsca jedna z podanych poniżej, nieprawidłowych operacji na liczbach typu REAL:

- $IN < 0$.
- IN nie jest liczbą.

W przeciwnym wypadku, sygnał wyjściowy nie jest przesyłany.



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie funkcji.
IN	Wartość stała lub adres zmiennej, dla której obliczany jest pierwiastek kwadratowy. Jeżeli parametr IN jest mniejszy od zera, funkcja nie powoduje przesłania sygnału Q.
ok	Sygnał wyjściowy wysyłany, jeżeli działanie zostało poprawnie wykonane, jego wynik mieści się w dopuszczalnym przedziale wartości i nie ma próby wykonania działania niewykonalnego.
Q	Wynik działania - część całkowita pierwiastka kwadratowego z liczby IN.

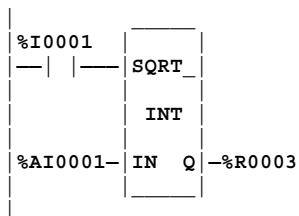
Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN		o	o	o	o		o	•	•	•	•†	
ok	•											•
Q		o	o	o	o		o	•	•	•		

- Typ może być wykorzystany jako dany parametr bloku funkcyjnego.
- o Typ może być wykorzystany wyłącznie jako parametr do operacji na danych typu INT. Nie można go wykorzystywać w przypadku operacji na danych typu DINT i REAL.
- † W operacjach wykonywanych na danych typu DINT, wartości stałe są ograniczone do przedziału <-32768, +32767>.

Przykład

W zamieszczonym poniżej przykładzie, zawsze gdy zmienna %I0001 jest równa 1, obliczany będzie pierwiastek kwadratowy z liczby %AI001, a wynik zapisywany będzie do zmiennej %R0003.



Funkcje trygonometryczne (SIN, COS, TAN, ASIN, ACOS, ATAN)

Funkcje SIN, COS i TAN służą do obliczania odpowiednio sinusa, cosinusa i tangensa dla zadanej liczby. Po doprowadzeniu sygnału do jednej z tych funkcji, obliczany jest sinus (cosinus lub tangens) dla liczby podanej jako parametr wejściowy IN, a wynik zapisywany jest w zmiennej wyjściowej Q. Zarówno parametr IN jak i Q są liczbami rzeczywistymi.

Funkcje ASIN, ACOS i ATAN służą do obliczania odpowiednio arcus sinus, arcus cosinus i arcus tangens dla zadanej liczby. Po doprowadzeniu sygnału do jednej z tych funkcji, obliczany jest sinus (cosinus lub tangens) dla liczby podanej jako parametr wejściowy IN, a wynik w radianach zapisywany jest w zmiennej wyjściowej Q. Zarówno parametr IN jak i Q są liczbami rzeczywistymi.

Funkcje SIN, COS i TAN dopuszczają szeroki zakres wartości wejściowych: $-2^{63} < IN < +2^{63}$, ($2^{63} \approx 9.22 \times 10^{18}$).

Funkcje ASIN i ACOS mają znacznie mniejszy zakres wartości wejściowych, a mianowicie: $-1 \leq IN \leq 1$. Po wprowadzeniu poprawnego parametru IN, funkcja ASIN_REAL zwraca przez parametr Q wynik spełniający podaną poniżej zależność:

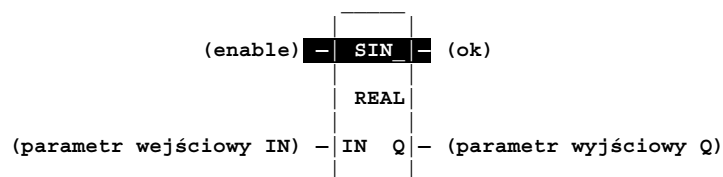
$$\text{ASIN (IN)} = -\frac{\pi}{2} \leq Q \leq \frac{\pi}{2}$$

ACOS_REAL zwraca przez parametr Q wynik spełniający podaną poniżej zależność:

$$\text{ACOS (IN)} = 0 \leq Q \leq \pi$$

Funkcja ATAN dopuszcza szeroki zakres wartości wejściowych, a mianowicie $-\infty \leq IN \leq +\infty$. Po wprowadzeniu poprawnego parametru IN, funkcja ATAN_REAL zwraca przez parametr Q wynik spełniający podaną poniżej zależność:

$$\text{ATAN (IN)} = -\frac{\pi}{2} \leq Q \leq \frac{\pi}{2}$$



Uwaga

Funkcje trygonometryczne mogą być realizowane **wyłącznie** w jednostkach centralnych 35x i 36x, oprogramowanie systemowe ver. 9 lub nowsza oraz we wszystkich wersjach jednostki centralnej CPU352.

Parametry

Parametr	Opis
enable	Sygnal wejściowy. Doprowadzenie tego sygnału powoduje wykonanie funkcji.
IN	Wartość stała lub adres zmiennej, będącej parametrem wejściowym funkcji.
ok	Sygnal wyjściowy wysyłany, jeżeli działanie zostało poprawnie wykonane, jego wynik mieści się w dopuszczalnym przedziale wartości, parametr wejściowy ma odpowiednią wartość i jest liczbą.
Q	Wynik działania - wartość obliczona przez funkcje trygonometryczną dla parametru wejściowego IN.

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN								•	•	•	•	
ok	•											•
Q								•	•	•		

- Dozwolony typ parametru.

Przykład

W zamieszczonym poniżej przykładzie obliczany jest cosinus dla wartości podanej w %R0001, a wynik zwracany jest poprzez zmienną %R0033.

```

ALW_ON
---] [-----|
                | COS_ | -
                | REAL |
                | IN  Q  |-%R0033
+3.141500|-----| -1.000000

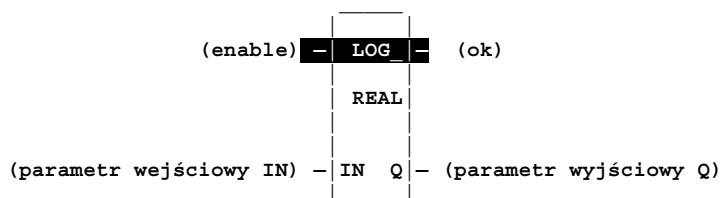
```

Funkcje logarytmiczne/wykładnicze (LOG, LN, EXP, EXPT)

Funkcje LOG, LN i EXP posiadają dwa parametry wejściowe i dwa parametry wyjściowe. Doprowadzenie sygnału do tej funkcji powoduje wykonanie odpowiedniej operacji logarytmicznej/ wykładniczej na parametrze wejściowym typu INT i zapisanie wyniku do parametru wyjściowego Q.

- Dla funkcji LOG, parametr wyjściowy Q jest równy logarytmowi o podstawie dziesiętnej z liczby IN.
- Dla funkcji LN, parametr wyjściowy Q jest równy logarytmowi naturalnemu z liczby IN.
- Dla funkcji EXP, liczba e podnoszona jest do potęgi zadanej parametrem IN, a wynik zapisywany jest do parametru wyjściowego Q.
- Funkcja EXPT podnosi wartość parametrem I1 do potęgi zadanej parametrem I2, a wynik zwracany jest poprzez parametr Q. (Funkcja EXPT posiada trzy parametry wejściowe i dwa parametry wyjściowe.)

Jeżeli parametr wejściowy IN jest liczbą i ma wartość dodatnią, na wyjście ok przesyłany jest sygnał.



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie funkcji.
IN	Wartość rzeczywista, będąca parametrem wejściowym funkcji.
ok	Sygnał wyjściowy wysyłany, jeżeli działanie zostało poprawnie wykonane, jego wynik mieści się w dopuszczalnym przedziale wartości, parametr wejściowy ma odpowiednią wartość i jest liczbą dodatnią.
Q	Wynik działania - wartość obliczona przez funkcje logarytmiczną/ wykładniczą dla parametru wejściowego IN.

Uwaga

Funkcje LOG, LN, EXP i EXPT mogą być realizowane **wyłącznie** w jednostkach centralnych 35x i 36x, oprogramowanie systemowe ver. 9 lub nowsza oraz we wszystkich jednostkach centralnych CPU352.

Uwaga

Jeżeli parametr wejściowy IN funkcji EXP ma wartość nieskończenie dużą, ujemną, funkcja ta zwraca wartość 0, zgodnie z oczekiwaniem. W jednostce centralnej CPU352, funkcja *nie* przesyła sygnału wyjściowego. We wszystkich pozostałych jednostkach centralnych serii 90-30, funkcja *przesyła* sygnał wyjściowy, nawet jeżeli wynik jest równy 0.

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN*								•	•	•	•	
ok	•											•
Q								•	•	•		

- * W przypadku funkcji EXPT, parametr wejściowy IN jest zastępowany przez parametry wejściowe I1 i I2.
- Dozwolony typ parametru.

Przykład

W zamieszczonym poniżej przykładzie, wynik podniesienia wartości %AI0001 do potęgi 2.5 jest zapisywany w %R0001.

```

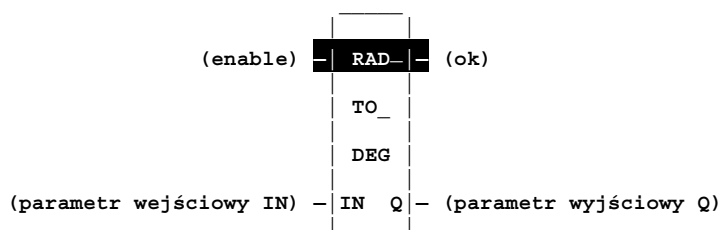
ALW_ON
—] [————— EXPT_ —
                REAL
                %AI0001— I1 Q —%R0001
                CONST — I2
                2.50000E+00

```


Konwersja miar kąta (RAD, DEG)

Po doprowadzeniu sygnału do tej funkcji, dla liczby podanej jako parametr wejściowy IN wykonywana jest odpowiednia konwersja wartości kąta ((RAD_TO_DEG lub DEG_TO_RAD), tzn. z wartości w radianach na wartość w stopniach lub odwrotnie, a wynik zapisywany jest w parametrze wyjściowym Q.

Jeżeli parametr IN jest liczbą, na wyjście ok przesyłany jest sygnał.



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie funkcji.
IN	Wartość rzeczywista, będącej parametrem wejściowym funkcji.
ok	Sygnał wyjściowy wysyłany, jeżeli wynik działania mieści się w dopuszczalnym przedziale wartości i parametr IN jest liczbą.
Q	Wynik działania - wartość parametru wejściowego IN wyrażona w innych jednostkach miary kąta.

Uwaga

Funkcje do konwersji miary kąta w radianach mogą być realizowane **wyłącznie** w jednostkach centralnych 35x i 36x, oprogramowanie systemowe wer. 9 lub nowsza oraz we wszystkich wersjach jednostki centralnej CPU352.

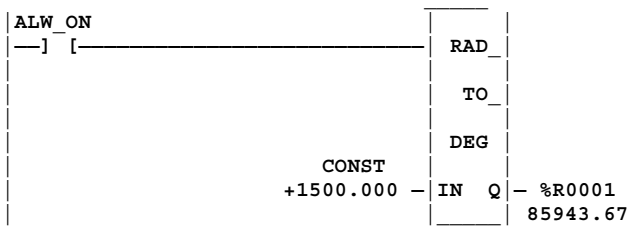
Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN								•	•	•	•	
ok	•											•
Q								•	•	•		

- Dozwolony typ parametru.

Przykład

W zamieszczonym poniżej przykładzie stała +1500 jest zamieniana na wartość w stopniach, a wynik przypisywany jest do zmiennej %R0001.



Opisane w tym rozdziale funkcje stosowane są do porównywania dwóch wartości. W rozdziale tym opisano następujące relacje:

Oznaczenie skrótowe	Element	Opis	Strona
EQ	Równy	Sprawdzenie, czy dwie liczby są równe.	7-2
NE	Różny	Sprawdzenie, czy dwie liczby mają różne wartości.	7-2
GT	Większy	Sprawdzenie, czy jedna liczba jest większa od drugiej.	7-2
GE	Większy lub równy	Sprawdzenie, czy jedna liczba jest większa lub równa drugiej.	7-2
LT	Mniejszy	Sprawdzenie, czy jedna liczba jest mniejsza od drugiej.	7-2
LE	Mniejszy lub równy	Sprawdzenie, czy jedna liczba jest mniejsza lub równa drugiej.	7-2
RANGE	Zakres pomiarowy	Sprawdzenie, czy liczba mieści się w zadanym przedziale (funkcja dostępna w jednostkach centralnych z oprogramowaniem systemowym ver. 4.5 lub nowsza).	7-4

Standardowe funkcje relacji (EQ, NE, GT, GE, LT, LE)

Doprowadzenie sygnału do tej funkcji powoduje porównanie parametru wejściowego I1 z parametrem wejściowym I2. Obydwa parametry wejściowe muszą być tego samego typu. Relacje matematyczne wykonują operacje na następujących typach danych:

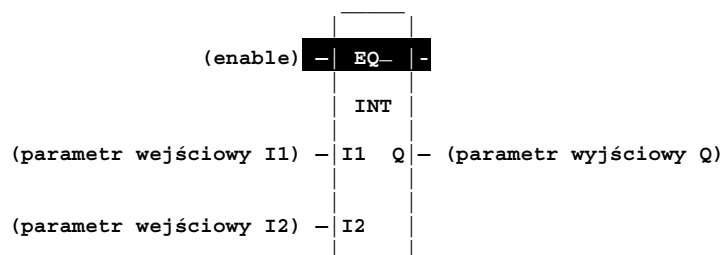
Typ	Opis
INT	Liczba całkowita ze znakiem (16 bitowa)
DINT	Liczba całkowita podwójnej precyzji ze znakiem (32 bitowa).
REAL	Liczba rzeczywista

Uwaga

Operacje na liczbach rzeczywistych można realizować wyłącznie w jednostkach centralnych 35x i 36x, oprogramowanie systemowe wer. 9 lub nowsza oraz we wszystkich wersjach jednostki centralnej CPU352. W przypadku, gdy funkcja relacji operująca na danych typu REAL zostanie pomyślnie wykonana, wartość bitu %S0020 ustawiana jest na 1. Jeżeli parametr wejściowy ma wartość NaN (Not a Number), bit ten jest zerowany. Parametrem wejściowym do bloku funkcyjnego Range nie może być liczba typu REAL.

Domyślny typ danych to liczba całkowita ze znakiem. W celu porównania liczb całkowitych ze znakiem, liczb całkowitych podwójnej precyzji ze znakiem lub liczb rzeczywistych należy wybrać innych typ danych po wybraniu relacji. W celu porównania danych innych typów lub danych o różnych typach, należy je uprzednio przekonwertować za pomocą odpowiedniej funkcji (proszę porównać z Rozdziałem 11 "Funkcje konwersji")

Jeżeli parametry wejściowe I1 i I2 spełniają daną relację, wartość parametru wyjściowego Q jest ustawiana na 1, w przeciwnym wypadku jest ona ustawiana na 0.



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie funkcji.
I1	Wartość stała lub adres zmiennej, będącej pierwszym parametrem sprawdzanej relacji. (Parametr I1 znajduje się z lewej strony działania matematycznego, jak na przykład w $I1 < I2$).
I2	Wartość stała lub adres zmiennej, będącej drugim parametrem sprawdzanej relacji. (Parametr I2 znajduje się z prawej strony działania matematycznego, jak na przykład w $I1 < I2$).
Q	Sygnał wyjściowy, wysyłany jeżeli parametry I1 i I2 spełniają daną relację.

Uwaga

Parametry I1 i I2 muszą być poprawnymi liczbami, tzn. muszą mieć wartości różne od NaN (Not a Number).

Rozszerzony opis

Element	Opis
EQ	Po doprowadzeniu sygnału do wejścia enable, jeżeli wartość I1 jest równa wartości I2, na wyjście Q wysyłany jest sygnał.
NE	Po doprowadzeniu sygnału do wejścia enable, jeżeli wartość I1 NIE jest równa wartości I2, na wyjście Q wysyłany jest sygnał.
GT	Po doprowadzeniu sygnału do wejścia enable, jeżeli wartość I1 jest większa od wartości I2, na wyjście Q wysyłany jest sygnał.
GE	Po doprowadzeniu sygnału do wejścia enable, jeżeli wartość I1 jest większa lub równa wartości I2, na wyjście Q wysyłany jest sygnał.
LT	Po doprowadzeniu sygnału do wejścia enable, jeżeli wartość I1 jest mniejsza od wartości I2, na wyjście Q wysyłany jest sygnał.
LE	Po doprowadzeniu sygnału do wejścia enable, jeżeli wartość I1 jest mniejsza lub równa wartości I2, na wyjście Q wysyłany jest sygnał.

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
I1		o	o	o	o		o	•	•	•	†	
I2		o	o	o	o		o	•	•	•	†	
Q	•											•

- Typ może być wykorzystany jako dany parametr bloku funkcyjnego.
- o Typ może być wykorzystany wyłącznie jako parametr do operacji na danych typu INT. Nie można go wykorzystywać w przypadku operacji na danych typu DINT i REAL.
- † W operacjach wykonywanych na danych typu DINT, wartości stałe są ograniczone do przedziału <-32768, +32767>.

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0001 na 1 powoduje porównanie dwóch liczb całkowitych podwójnej precyzji ze znakiem PWR_MDE i BIN_FUL. Jeżeli zmienna PWR_MDE jest mniejsza lub równa zmiennej BIN_FUL, następuje ustawienie zmiennej przekaźnika %Q0002 na 1.



Range (INT, DINT, WORD)

Funkcja RANGE służy do sprawdzania, czy dana liczba mieści się w przedziale wyznaczonym przez dwie inne liczby.

Uwaga

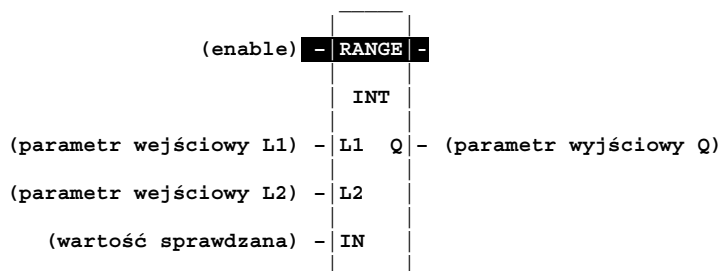
Funkcja ta dostępna jest wyłącznie w jednostkach centralnych z oprogramowaniem systemowym ver. 4.41 lub nowsza.

Funkcja RANGE operuje na następujących typach danych:

Typ	Opis
INT	Liczba całkowita ze znakiem (16 bitowa)
DINT	Liczba całkowita podwójnej precyzji ze znakiem (32 bitowa).
WORD	Dane typu Word (Słowo).

Domyślnym typem danych (wybieranym automatycznie przez oprogramowanie) jest INT, jednakże może on zostać zmieniony po dokonaniu wyboru bloku funkcyjnego i wprowadzeniu go do szczebla drabiny logicznej programu sterującego. Więcej informacji o typach danych podano w Rozdziale 2 w punkcie "Organizacja programu. Typy danych i typy zmiennych."

Po doprowadzeniu sygnału do tego bloku funkcyjnego, następuje sprawdzenie, czy parametr wejściowy IN mieści się w przedziale określonym przez wartości graniczne L1 i L2. Jeżeli jego wartość mieści się w przedziale domkniętym wyznaczonym przez L1 i L2, na wyjście Q przesyłany jest sygnał wyjściowy. W przeciwnym wypadku, sygnał wyjściowy nie jest przesyłany.



Uwaga

Parametry L1 i L2 określają granice przedziału. Nie jest sprawdzane, który z tych parametrów określa lewą stronę przedziału, a który prawą. Tak więc jeżeli żądany przedział wynosi od 0 do 100, wartość 0 powinna być przypisana do L1, a wartość 100 do L2.

Parametry

Parametr	Opis
enable	Sygnal wejściowy. Doprowadzenie tego sygnału powoduje wykonanie funkcji.
L1	Lewy koniec przedziału.
L2	Prawy koniec przedziału.
IN	Wartość stała lub adres zmiennej, która ma się mieścić w określonym przedziale.
Q	Sygnal wyjściowy, wysyłany jeżeli parametr IN znajduje się wewnątrz przedziału domkniętego, określonego przez L1 i L2.

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
L1		o	o	o	o		o	•	•	•	•‡	
L2		o	o	o	o		o	•	•	•	•‡	
IN		o	o	o	o		o	•	•	•		
Q	•											•

- Typ może być wykorzystany jako dany parametr bloku funkcyjnego.
- o Typ może być wykorzystany wyłącznie jako parametr do operacji na danych typu INT lub Word. Nie można go wykorzystywać w przypadku operacji na danych typu DINT.
- ‡ W operacjach wykonywanych na danych typu DINT, wartości stałe są ograniczone do przedziału <-32768, +32767>.

Przykład 1

W zamieszczonym poniżej przykładzie sprawdzane jest, czy wartość %AI0001 mieści się w przedziale określonym przez stałe 0 i 100.

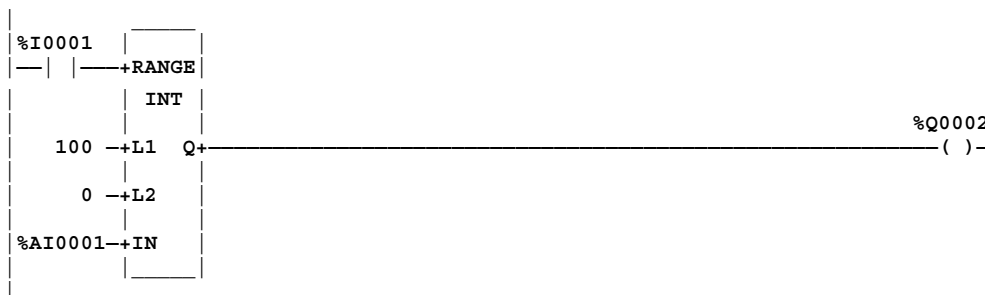


Tabela prawdziwości dla funkcji RANGE				
Sygnal Enable	Wartość L1 stała	Wartość L2 stała	Wartość IN %AI0001	Sygnal Q %Q0001
ON	100	0	< 0	OFF
ON	100	0	0 — 100	ON
ON	100	0	> 100	OFF
OFF	100	0	Nie dotyczy	OFF

Przykład 2

W zamieszczonym poniżej przykładzie, sprawdzane jest, czy wartość %AI0001 mieści się w przedziale określonym przez dwie wartości zapisane w rejestrach.

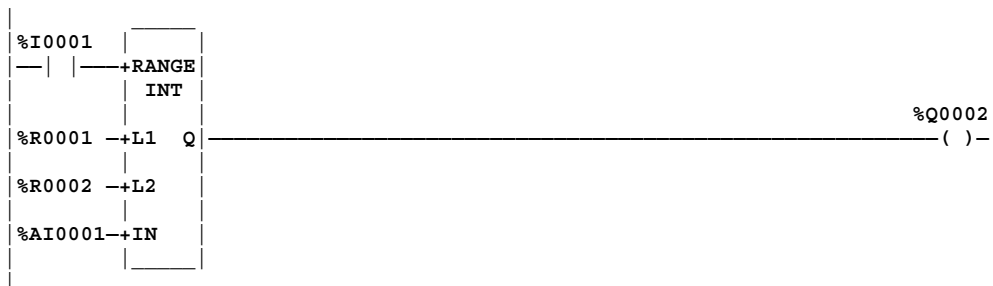


Tabela prawdziwości dla funkcji RANGE				
Sygnal Enable	Wartość L1 %R0001	Wartość L2 %R0002	Wartość IN %AI0001	Sygnal Q %Q0001
ON	500	0	< 0	OFF
ON	500	0	0 — 500	ON
ON	500	0	> 500	OFF
OFF	500	0	Nie dotyczy	OFF

Rozdział 8

Funkcje do operacji na ciągach bitów

Bloki funkcyjne z tej grupy wykonują operacje logiczne na ciągach bitów. Funkcje AND, OR, XOR i NOT wykonują operacje na pojedynczym słowie. Pozostałe funkcje z tej grupy mogą wykonywać działania na ciągu słów, długość takiego ciągu nie może przekraczać 256 słów. Wszystkie funkcje do operacji bitowych wymagają danych typu WORD.

Dane są wprowadzicie pogrupowane w 16-bitowe słowa, lecz traktowane jako nieprzerwany ciąg bitów, z pierwszym bitem pierwszego słowa stanowiącym bit najbardziej znaczący (MSB), i ostatnim bitem ostatniego słowa stanowiącym bit najmniej znaczący (LSB). Przykładowo, jeśli funkcja ma wykonać operację na trzech słowach o adresie początkowym %R0100, to wykona ją na 48 kolejnych bitach, traktowanych w sposób pokazany na poniższym schemacie.

%R0100	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	← bit 1 (LSB)
%R0101	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	
%R0102	48	47	46	45	44	43	42	41	40	39	38	37	36	35	34	33	

↑
(MSB)

Uwaga

Zachodzące na siebie zakresy adresów parametrów wejściowych i wyjściowych w przypadku funkcji operujących na ciągach wielu słów bitowych mogą być przyczyną uzyskania nieoczekiwanych wyników.

W niniejszym rozdziale opisano następujące funkcje do operacji bitowych:

Oznaczenie skrótowe	Element	Opis	Strona
AND	Logiczne AND	Jeżeli zarówno bit parametru wejściowego I1 jak i odpowiadający mu bit parametru wejściowego I2 mają wartość 1, wartość odpowiedniego bitu parametru wyjściowego Q jest ustawiana na 1.	8-3
OR	Logiczne OR	Jeżeli bit parametru wejściowego I1 ma wartość 1 i/lub odpowiadający mu bit parametru wejściowego I2 ma wartość 1, wartość odpowiedniego bitu parametru wyjściowego Q jest ustawiana na 1.	8-3
XOR	Alternatywa wyłączająca	Jeżeli bit parametru wejściowego I1 i odpowiadający mu bit parametru wejściowego I2 mają różne wartości, wartość odpowiedniego bitu parametru wyjściowego Q jest ustawiana na 1.	8-5
NOT	Negacja logiczna	Każdy bit w parametrze wyjściowym Q jest ustawiany na wartość przeciwną w stosunku do wartości odpowiedniego bitu parametru I1.	8-7
SHL	Przesunięcie w lewo	Przesunięcie wszystkich bitów jednego słowa bitowego lub ciągu kilku słów bitowych w lewo, o wyszczególnioną liczbę miejsc.	8-7
SHR	Przesunięcie w prawo	Przesunięcie wszystkich bitów jednego słowa bitowego lub ciągu kilku słów bitowych w prawo, o wyszczególnioną liczbę miejsc.	8-7
ROL	Przesunięcie w lewo w obiegu zamkniętym	Przesunięcie wszystkich bitów jednego słowa bitowego lub ciągu kilku słów bitowych w lewo, o wyszczególnioną liczbę miejsc, przy czym najbardziej znaczące bity (z lewej strony słowa) "wypchnięte" ze słowa bitowego zostają wpisane na puste miejsca z prawej strony słowa.	8-9
ROR	Przesunięcie w prawo w obiegu zamkniętym	Przesunięcie wszystkich bitów jednego słowa bitowego lub ciągu kilku słów bitowych w prawo, o wyszczególnioną liczbę miejsc, przy czym najmniej znaczące bity (z prawej strony słowa) "wypchnięte" ze słowa bitowego zostają wpisane na puste miejsca z lewej strony słowa.	8-57
BTST	Sprawdzanie wartości pojedynczego bitu	Sprawdzanie wartości (0 lub 1) jednego z bitów słowa bitowego.	8-12
BSET	Ustawianie wartości pojedynczego bitu na 1	Ustawianie wartości danego bitu w słowie bitowym na 1.	8-14
BCLR	Ustawianie wartości pojedynczego bitu na 0	Ustawienie wartości określonego bitu w ciągu bitowym na 0.	8-14
BPOS	Lokalizowanie pierwszego bitu o wartości 1	Przeszukiwanie słowa bitowego (lub ciągu słów) do napotkania pierwszego bitu o wartości równej 1.	8-16
MSKCMP	Porównanie z maskowaniem	Porównanie kolejnych bitów dwóch ciągów bitów, z możliwością maskowania wybranych bitów (funkcja dostępna w jednostkach centralnych z oprogramowaniem systemowym ver. 4.5 lub nowsza).	8-18

AND i OR (WORD)

Za każdym razem, po doprowadzeniu sygnału, funkcje AND i OR porównują każdy bit parametru I1 z odpowiednim bitem parametru I2, począwszy od najmniej znaczących bitów.

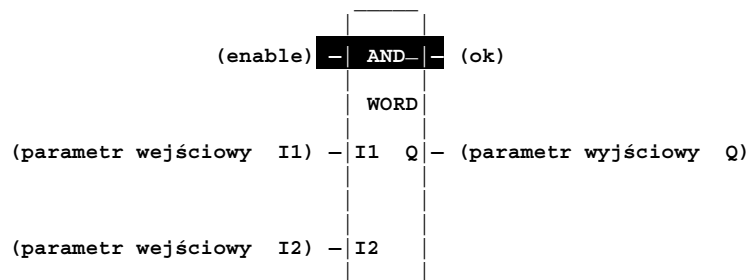
W przypadku funkcji AND, wartość każdego bitu parametru wyjściowego Q jest ustawiana na 1, jeśli odpowiednie bity pierwszego i drugiego parametru wejściowego (tzn. słów I1 oraz I2) mają wartość 1. Jeśli jeden lub obydwa bity mają wartość 0, to wartość odpowiedniego bitu słowa Q ustawiana jest na 0.

Funkcja AND jest użyteczna do tworzenia masek, umożliwiających zmianę stanu (np. wyzerowanie) wyszczególnionych bitów pamięci lub wyzerowanie zadanego obszaru pamięci poprzez koniunkcję ze słowem zawierającym same zera. Ciągi bitów I1 i I2 mogą zajmować ten sam obszar pamięci.

W przypadku funkcji OR, wartość każdego bitu parametru wyjściowego Q jest ustawiana na 1, jeśli jeden lub obydwa odpowiednie bity pierwszego i drugiego parametru wejściowego (tzn. słów I1 oraz I2) mają wartość 1.

Funkcja OR jest stosowana np. do sterowania stanem wielu wyjść za pomocą jednej struktury logicznej. Blok funkcyjny OR jest równoważny układowi dwóch przełączników połączonych równolegle, zwielokrotnionego tyle razy, ile bitów zawierają porównywane słowa. Może on być wykorzystywany do bezpośredniego sterowania lampkami sygnalizacyjnymi w zależności od stanów wejść lub ustanowienia wyższego priorytetu błyskania nad zapaleniem.

Sygnal wyjściowy jest wysyłany zawsze po doprowadzeniu sygnału wejściowego.



Parametry

Parametr	Opis
enable	Sygnal wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
I1	Wartość stała lub adres zmiennej, stanowiącej pierwsze słowo, na którym ma być wykonywana operacja logiczna.
I2	Wartość stała lub adres zmiennej, stanowiącej drugie słowo, na którym ma być wykonywana operacja logiczna.
ok	Sygnal wyjściowy wysyłany po doprowadzeniu do bloku funkcyjnego sygnału wejściowego.
Q	Wynik działania.

Dopuszczalne typy parametrów

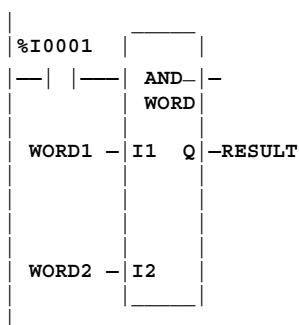
Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
I1		•	•	•	•	•	•	•	•	•	•	
I2		•	•	•	•	•	•	•	•	•	•	
ok	•											•
Q		•	•	•	•	•†	•	•	•	•		

• Dozwolony typ parametru.

† Można stosować wyłącznie zmienne systemowe typu %SA, %SB i %SC. Nie można korzystać ze zmiennych typu %S.

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0001 na 1 powoduje wykonanie operacji logicznej na dwóch 16 bitowych parametrach o nazwach pomocniczych Word1 i Word2. Wynik działania funkcji AND zapisywany jest do parametru wyjściowego RESULT.



WORD1	0	0	0	1	1	1	1	1	1	1	0	0	1	0	0	0
WORD2	1	1	0	1	1	1	0	0	0	0	0	0	1	1	1	1
RESULT	0	0	0	1	1	1	0	0	0	0	0	0	1	0	0	0

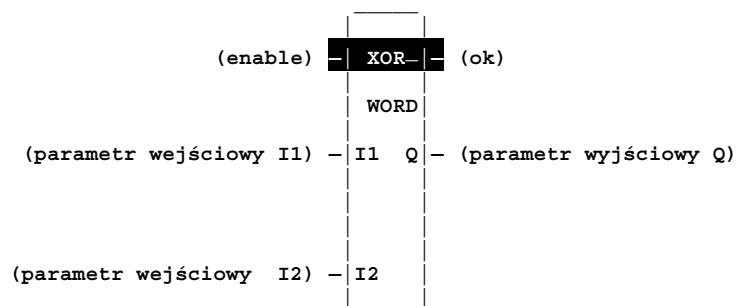
XOR (WORD)

Funkcja XOR umożliwia wykonanie operacji alternatywy wyłączającej dwóch słów bitowych, będących parametrami wejściowymi I1 i I2 funkcji. Wynikiem operacji jest bit o wartości 1, jeśli odpowiadające sobie bity mają różne wartości.

Po doprowadzeniu sygnału do bloku funkcyjnego, funkcja wykonuje tę operację dla każdej pary bitów, z których pierwszy jest kolejnym bitem słowa I1, a drugi odpowiednim bitem, słowa I2, począwszy od najmniej znaczących bitów w każdym ze słów. Wynikiem operacji na dwóch sprawdzanych bitach jest bit o wartości 1, jeśli jeden ze sprawdzanych bitów ma wartość 1, a drugi z nich ma wartość 0. Sygnał wyjściowy jest przesyłany przez blok funkcyjny XOR zawsze po doprowadzeniu sygnału wejściowego.

Jeśli słowa I2 oraz Q mają ten sam adres początkowy, bit słowa I1 o wartości 1 spowoduje migotanie (cykliczną zmianę wartości pomiędzy 1 a 0) odpowiedniego bitu słowa I2, zgodnie z częstotliwością wykonywania cykli pracy przez sterownik. Migotanie z mniejszą częstotliwością można wywołać, podając sygnał na wejście bloku funkcyjnego z częstotliwością równą podwójnej żądanej częstotliwości migotania

Funkcja XOR może być wykorzystana do szybkiego porównania dwóch ciągów bitów lub do przełączania grupy bitów, z ustawianiem wartości 1 w co drugim cyklu pracy sterownika



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
I1	Wartość stała lub adres zmiennej będącej pierwszym parametrem wykonywanej funkcji XOR.
I2	Wartość stała lub adres zmiennej będącej drugim parametrem wykonywanej funkcji XOR.
ok	Sygnał wyjściowy wysyłany po doprowadzeniu sygnału wejściowego.
Q	Wynik operacji logicznej XOR na parametrach I1 i I2.

Dopuszczalne typy parametrów

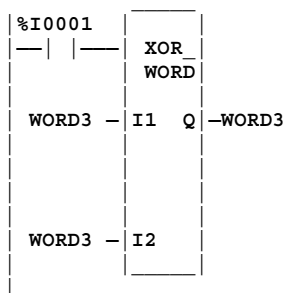
Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
I1		•	•	•	•	•	•	•	•	•	•	
I2		•	•	•	•	•	•	•	•	•	•	
Ok	•											•
Q		•	•	•	•	•†	•	•	•	•		

• Dozwolony typ parametru.

† Można stosować wyłącznie zmienne systemowe typu %SA, %SB i %SC. Nie można korzystać ze zmiennych typu %S.

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0001 na 1 powoduje ustawienie na zero wszystkich bitów parametru wejściowego o nazwie pomocniczej WORD3.

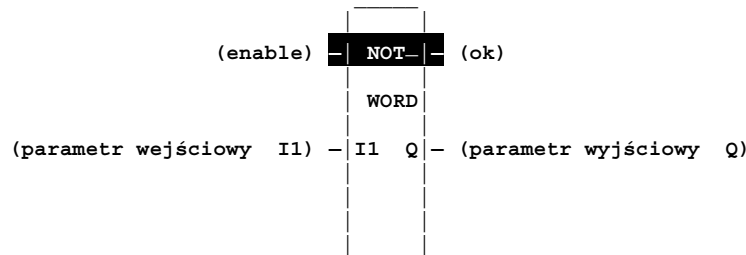


I1 (WORD3)	0	0	0	1	1	1	1	1	1	1	0	0	1	0	0	0
I2 (WORD3)	0	0	0	1	1	1	1	1	1	1	0	0	1	0	0	0
Q (WORD3)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

NOT (WORD)

Funkcja negacji NOT jest wykorzystywana do zmiany stanu każdego bitu słowa, które jest parametrem wejściowym I1 bloku funkcyjnego, na przeciwny.

Operacja ta wykonywana jest zawsze po doprowadzeniu sygnału wejściowego. Wynikiem operacji jest słowo bitowe (parametr wyjściowy Q), stanowiące logiczną negację słowa I1. Sygnał wyjściowy jest wysyłany zawsze po doprowadzeniu sygnału wejściowego.



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
I1	Wartość stała lub adres zmiennej będącej parametrem wykonywanej funkcji NOT.
ok	Sygnał wyjściowy wysyłany po doprowadzeniu sygnału wejściowego.
Q	Wynik działania - negacja słowa I1.

Dopuszczalne typy parametrów

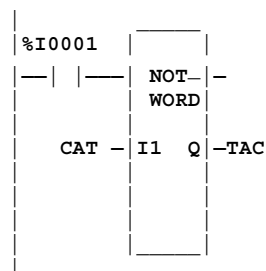
Parametr	Sygnał	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
I1		•	•	•	•	•	•	•	•	•	•	
ok	•											•
Q		•	•	•	•	•†	•	•	•	•		

• Dozwolony typ parametru.

† Można stosować wyłącznie zmienne systemowe typu %SA, %SB i %SC. Nie można korzystać ze zmiennych typu %S.

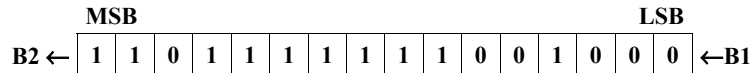
Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0001 na 1 powoduje ustawienie wszystkich bitów zmiennej TAC na wartość przeciwną w stosunku do odpowiednich bitów zmiennej CAT.

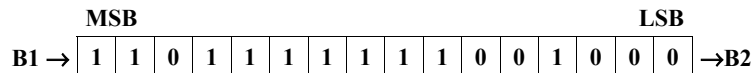


SHL i SHR (WORD)

Funkcję SHL można wykorzystać do przesunięcia wszystkich bitów jednego słowa bitowego lub ciągu kilku słów bitowych w lewo, o wyszczególnioną liczbę miejsc. Starsze bity (z lewej strony słowa) zostają "wypchnięte" ze słowa bitowego. Na puste miejsca zostają wpisane zadane wartości.



Funkcję SHR można wykorzystać do przesunięcia wszystkich bitów jednego słowa bitowego lub ciągu kilku słów bitowych w prawo, o wyszczególnioną liczbę miejsc. Młodsze bity (z prawej strony słowa) zostają "wypchnięte" ze słowa bitowego. Na puste miejsca zostają wpisane zadane wartości.



Każda z powyższych funkcji może wykonywać przesunięcie w zakresie od 1 do 256 słów bitowych.

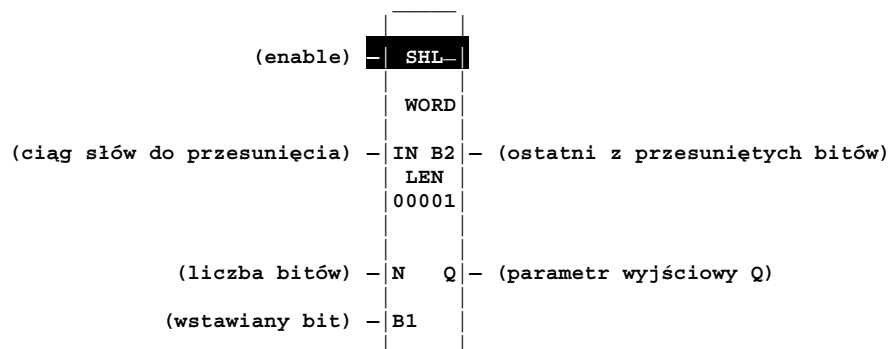
Jeżeli liczba miejsc, o jaką mają być przesunięte bity danego słowa (N) jest większa od liczby bitów zawartych w zadanym ciągu (LEN)*16 lub jeżeli liczba ta jest równa 0, do ciągu wyjściowego (Q) kopiowany jest parametr wejściowy B1, a bit wejściowy kopiowany jest do parametru wyjściowego B2, zawierającego wartość ostatniego bitu, który wyszedł poza zakres słowa po dokonaniu operacji przesunięcia. Jeżeli liczba miejsc, o jaką mają być przesunięte bity danego słowa jest równa zero, przesunięcie nie zostaje wykonane, ciąg wejściowy jest kopiowany do ciągu wyjściowego, a bit wejściowy B1 jest kopiowany do parametru wyjściowego B2.

W puste miejsca powstałe po przesunięciu zawartości słowa (ciągu słów) wpisane zostają bity o wartości zadanej poprzez parametr B1. Jeżeli liczba miejsc, o którą mają być przesunięte bity danego słowa jest większa od 1, każdy z bitów ma ustawianą taką samą wartość (0 lub 1). Jako parametr ten można wykorzystać:

- Zerojedynekowe wyjście bloku funkcyjnego.
- Sygnał wyjściowy przekaźnika o przypisanej zmiennej systemowej ALW_ON; spowoduje to, że w puste miejsca zostaną wpisane same jedynki.
- Sygnał wyjściowy przekaźnika o przypisanej zmiennej systemowej ALW_OFF; spowoduje to, że w puste miejsca zostaną wpisane same zera.

Funkcje SHL i SHR zawsze przesyłają sygnał wyjściowy, o ile liczba miejsc, o jaką mają być przesunięte bity danego słowa jest większa od zera.

Wynik, uzyskany po wykonaniu operacji przesunięcia jest kopiowany w miejsce wyszczególnione przez parametr Q. Jeżeli mają być przesunięte bity w parametrze wejściowym, parametr wyjściowy Q musi mieć taki sam adres jak parametr wejściowy IN. Łańcuch bitowy z przesuniętymi bitami zapisywany jest za każdym razem po doprowadzeniu sygnału. Parametr B2 zawiera ostatni bit, który wyszedł poza zakres słowa po dokonaniu operacji przesunięcia. Jeśli na przykład liczba bitów, o którą miały zostać przesunięte bity słowa (ciągu słów) wynosi 4, parametr B2 będzie zawierał czwarty z bitów, które zostały wypchnięte poza słowo.



Parametry

Parametr	Opis
enable	Sygnal wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
IN	Adres pierwszego słowa ciągu słów, którego bity mają zostać przesunięte.
N	Liczba miejsc (bitów), o które mają zostać przesunięte bity danego słowa (ciągu słów).
B1	Wartość bitu (bitów), które mają zostać wstawione w puste miejsca słowa, powstałe po przesunięciu jego zawartości.
B2	Wartość ostatniego bitu, który wyszedł poza zakres słowa po dokonaniu operacji przesunięcia.
Q	Adres pierwszego słowa ciągu słów, otrzymanego po przesunięciu bitów słowa adresowanego przez parametr IN.
LEN	Liczba słów bitowych obszaru, w obrębie którego ma zostać dokonane przesunięcie bitów.

Dopuszczalne typy parametrów

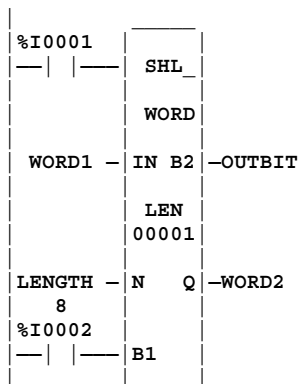
Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN		•	•	•	•	•	•	•	•	•		
N		•	•	•	•		•	•	•	•	•	
B1	•											
B2	•											•
Q		•	•	•	•	•†	•	•	•	•		

• Dozwolony typ parametru.

† Można stosować wyłącznie zmienne systemowe typu %SA, %SB i %SC. Nie można korzystać ze zmiennych typu %S.

Przykład

W zamieszczonym poniżej przykładzie, po każdej zmianie wartości zmiennej %I0001 na 1 następuje zapisanie do zmiennej o nazwie WORD2 wyniku wykonania operacji przesunięcia ciągu bitów WORD1 w lewo, o liczbę określoną przez zmienną LENGTH. Puste miejsca powstałe na początku wyjściowego ciągu bitów są ustawiane na wartość określoną zmienną %I0002.



ROL i ROR (WORD)

Funkcję ROL można wykorzystać do przesunięcia wszystkich bitów jednego słowa bitowego lub ciągu kilku słów bitowych w lewo, o wyszczególnioną liczbę miejsc. Najbardziej znaczące bity (z lewej strony słowa), "wypchnięte" ze słowa bitowego zostają wpisane na puste miejsca z prawej strony słowa.

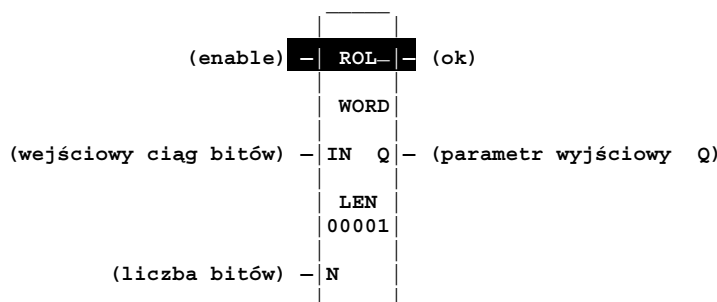
Funkcja ROR jest analogiczna do funkcji SHL, lecz powoduje przesunięcie wszystkich bitów jednego słowa bitowego lub ciągu kilku słów bitowych w prawo. Najmniej znaczące bity (z prawej strony słowa), "wypchnięte" ze słowa bitowego, zostają wpisane na puste miejsca z lewej strony słowa.

Każda z powyższych funkcji może wykonywać przesunięcie w zakresie od 1 do 256 słów bitowych.

Jeżeli liczba miejsc (N) o jaką mają być przesunięte bity danego słowa jest równa Q, następuje przesunięcie słowa wejściowego (IN) do parametru wyjściowego (Q) i wysyłany jest sygnał wyjściowy. Jeżeli liczba miejsc (N) o jaką mają być przesunięte bity danego słowa jest większa od (LEN)*16 to blok nie wykonuje żadnego przesunięcia i nie wysyła sygnału potwierdzenia OK.

Poza tymi przypadkami nie ma ograniczeń dotyczących przesyłania sygnału wyjściowego.

Wynik funkcji zapisywany jest do ciągu wyjściowego Q. Jeżeli należy przesunąć bity w łańcuchu będącym parametrem wejściowym, adres parametru wyjściowego Q musi być taki sam jak adres parametru wejściowego IN. Ciąg bitowy z przesuniętymi bitami zapisywany jest za każdym razem po doprowadzeniu sygnału.



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
IN	Adres pierwszego słowa ciągu słów, którego bity mają zostać przesunięte.
N	Liczba miejsc (bitów), o które mają zostać przesunięte bity danego słowa (ciągu słów).
ok	Sygnał wyjściowy, który pojawia się po dokonaniu rotacji.
Q	Adres pierwszego słowa lub ciągu słów, otrzymanego po przesunięciu bitów słowa adresowanego przez parametr IN.
LEN	Liczba słów bitowych obszaru, w obrębie którego ma zostać dokonana rotacja bitów.

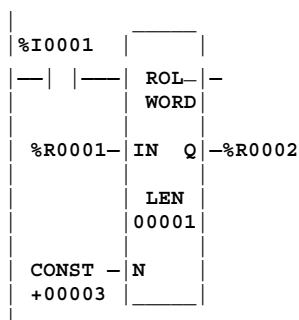
Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN		•	•	•	•	•	•	•	•	•		
N		•	•	•	•		•	•	•	•	•	
ok	•											•
Q		•	•	•	•	•†	•	•	•	•		

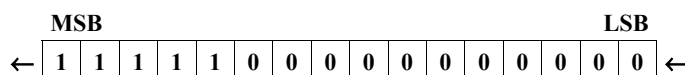
- Dozwolony typ parametru.
- † Można stosować wyłącznie zmienne systemowe typu %SA, %SB i %SC. Nie można korzystać ze zmiennych typu %S.

Przykład

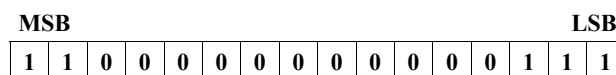
W zamieszczonym poniżej przykładzie, po każdej zmianie wartości zmiennej %I0001 na 1 następuje zapisanie do wyjściowego ciągu bitów %R0002 wyniku wykonania operacji przesunięcia ciągu bitów %R0001 w prawo, w obiegu zamkniętym, o 3 bity. Wykonanie tej funkcji nie powoduje zmiany wejściowego ciągu bitów %R0001. Jeżeli adresy parametru IN i Q będą takie same, wejściowy ciąg bitów zostanie zmieniony.



%R0001:



%R0002 (po ustawieniu wartości zmiennej %I0001 na 1):

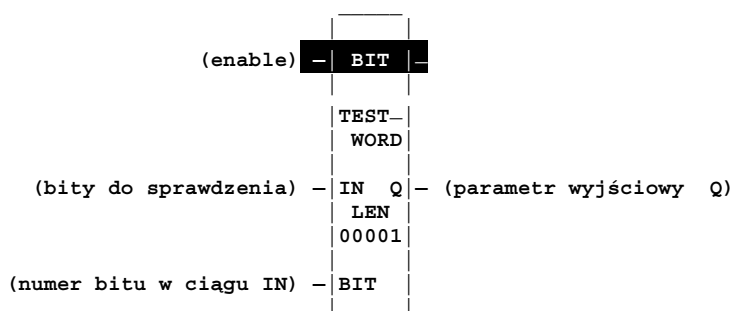


BTST (WORD)

Funkcja BTST jest stosowana do określenia wartości (0 lub 1) jednego z bitów słowa bitowego. Wynik testu zostaje zapisany jako parametr Q.

Jego wartość jest ustawiana zgodnie z wartością wyszczególnionego bitu sprawdzanego słowa. Parametr BIT może być również zmienną, co umożliwia sprawdzenie różnych bitów danego ciągu bitowego za pomocą tego samego bloku funkcyjnego. Jeśli wyszczególniony numer bitu (BIT) przekracza dopuszczalny zakres ($1 \leq \text{BIT} \leq [16 * \text{LEN}]$), wartość parametru wyjściowego jest ustawiana na 0.

Powyższa funkcja może dokonać sprawdzenia wartości bitu w zakresie od 1 do 256 słów bitowych, czyli z ciągu liczącego maksymalnie $16 * 256$ bitów.



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
IN	Adres pierwszego słowa ciągu słów, na którym ma zostać wykonana operacja.
BIT	Numer bitu słowa IN, którego wartość ma zostać sprawdzona. Zakres wartości to $1 \leq \text{BIT} \leq 16 * \text{LEN}$.
Q	Wartość parametru Q zostaje ustawiona na 1, gdy wartość testowanego bitu również wynosi 1.
LEN	Liczba słów bitowych ciągu słów, z którego wybierany jest bit do sprawdzania wartości (maksymalnie 256).

Uwaga

W przypadku korzystania z funkcji Bit Test, Bit Set, Bit Clear lub Bit Pos, bity numerowane są od 1 do 16, a NIE od 0 do 15.

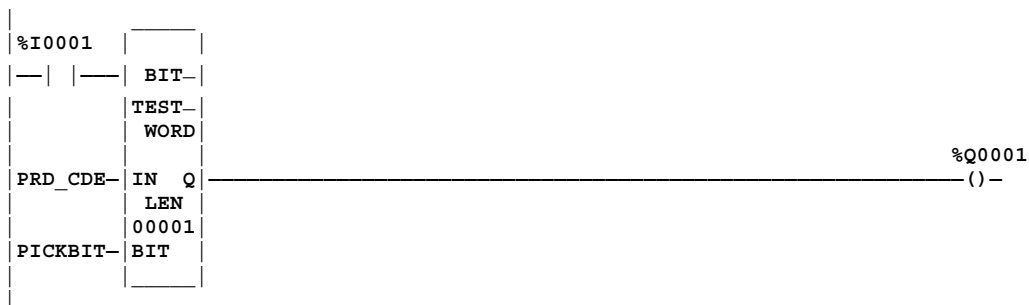
Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN		•	•	•	•	•	•	•	•	•		
BIT		•	•	•	•		•	•	•	•	•	
Q	•											•

- Dozwolony typ parametru.

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0001 na 1 powoduje sprawdzenie bitu, którego położenie określone jest zmienną PICKBIT. Bit ten należy do ciągu wejściowego PRD_CDE. Jeżeli jego wartość jest równa 1, następuje wysłanie sygnału wyjściowego Q oraz ustawienie zmiennej przekaźnika %Q0001 na 1.

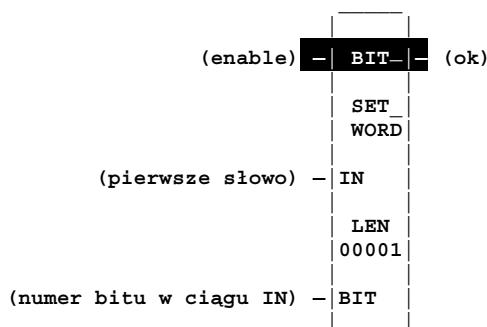


BSET i BCLR (WORD)

Funkcja BSET jest stosowana do ustawiania na 1 wartości określonego bitu ciągu bitowego. Funkcja BCLR jest stosowana do zerowania wartości określonego bitu ciągu bitowego.

Każde doprowadzenie sygnału powoduje ustawienie wartości określonego bitu na 1 (funkcja BSET) lub wyzerowanie go (funkcja BCLR). Parametr BIT może być również zmienną, co umożliwia ustawienie różnych bitów danego słowa bitowego za pomocą tego samego bloku funkcyjnego.

Powyższa funkcja może dokonać sprawdzenia wartości bitu w zakresie od 1 do 256 słów bitowych, czyli z ciągu liczącego maksymalnie $16 \cdot 256$ bitów. Jeśli wyszczególniony numer bitu (BIT) przekracza dopuszczalny zakres ($1 \leq \text{BIT} \leq [16 \cdot \text{LEN}]$), sygnał wyjściowy nie jest przesyłany.



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
IN	Adres pierwszego słowa ciągu słów, na którym ma zostać wykonana operacja.
BIT	Numer bitu słowa IN, którego wartość ma zostać ustawiona na 1 lub 0. Zakres wartości to $1 \leq \text{BIT} \leq 16 \cdot \text{LEN}$.
ok	Sygnał wyjściowy wysyłany po doprowadzeniu sygnału wejściowego.
LEN	Liczba słów bitowych ciągu słów, z którego wybierany jest bit, którego wartość ma zostać zmieniona (maksymalnie 256).

Uwaga

W przypadku korzystania z funkcji Bit Test, Bit Set, Bit Clear lub Bit Pos, bity numerowane są od 1 do 16, a **NIE** od 0 do 15.

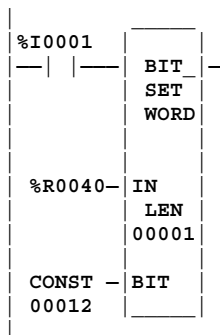
Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN		•	•	•	•	†	•	•	•	•		
BIT		•	•	•	•		•	•	•	•	•	
ok	•											•

- Dozwolony typ parametru.
- † Można stosować wyłącznie zmienne systemowe typu %SA, %SB i %SC. Nie można korzystać ze zmiennych typu %S.

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej wejściowej %I0001 na 1 powoduje ustawienie wartości 12 bitu na 1, w ciągu bitów, którego początek wskazywany jest przez adres %R0040.



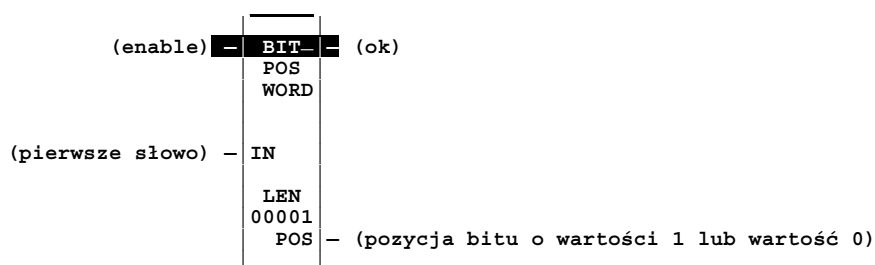
BPOS (WORD)

Funkcja BPOS przeszukuje słowo bitowe (lub ciąg słów) do napotkania pierwszego bitu o wartości równej 1.

Operacja ta jest dokonywana po doprowadzeniu sygnału wejściowego, rozpoczynając od najmłodszego bitu pierwszego słowa, adresowanego przez parametr IN. Blok funkcyjny kończy działanie, gdy znaleziony zostaje bit o wartości 1 lub gdy w całym przeszukiwanym obszarze wszystkie bity mają wartość równą 0.

Pozycja zlokalizowanego bitu jest udostępniana przez parametr POS. Jeśli wszystkie bity w przeszukiwanym ciągu słów posiadają wartość 0, wartość tego parametru jest ustawiana na 0.

Powyższa funkcja może operować na ciągu liczącym od 1 do 256 słów bitowych, czyli na ciągu liczącym maksymalnie 16*256 bitów. Sygnał wyjściowy jest wysyłany zawsze, po doprowadzeniu sygnału wejściowego.



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału do bloku funkcyjnego powoduje rozpoczęcie szukania.
IN	Adres pierwszego słowa ciągu słów, na którym ma zostać wykonana operacja.
ok	Sygnał wyjściowy wysyłany po doprowadzeniu sygnału wejściowego.
POS	Pozycja pierwszego znalezionej niezerowego bitu przeszukiwanego słowa lub 0 w przypadku, gdy w przeszukiwanym obszarze występują tylko wyzerowane bity.
LEN	Liczba słów bitowych ciągu słów, z którego wybierany jest bit, którego wartość ma zostać zmieniona (maksymalnie 256).

Uwaga

W przypadku korzystania z funkcji Bit Test, Bit Set, Bit Clear lub Bit Pos, bity numerowane są od 1 do 16, a NIE od 0 do 15.

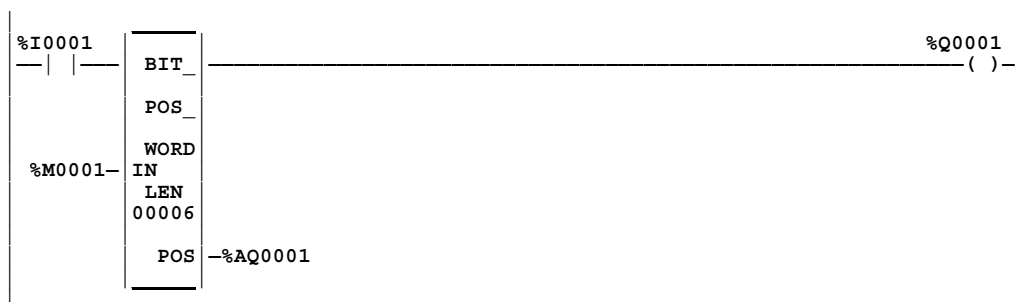
Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
Enable	•											
IN		•	•	•	•	•	•	•	•	•		
POS		•	•	•	•		•	•	•	•		
Ok	•											•

- Dozwolony typ parametru.

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0001 na 1 powoduje rozpoczęcie przeszukiwania ciągu bitów, począwszy od adresu wskazywanego przez %M0001, aż do momentu, kiedy odnaleziony zostanie bit o wartości 1 lub przeszukanych zostanie 6 słów. Zmienna przekaźnika %Q0001 jest ustawiana na 1. Jeżeli znaleziony zostanie bit o wartości 1, jego położenie w ciągu bitów jest zapisywane do zmiennej %AQ001. Jeżeli wartość zmiennej %I0001 jest równa 1, bit %M0001 jest równy 0, a bit %M0002 jest równy 1, to do zmiennej %AQ001 zapisywana jest wartość 2.



MSKCMP (WORD, DWORD)

Funkcja MSKCMP (*dostępna w jednostkach centralnych z oprogramowaniem systemowym ver. 4.41 lub nowszą*) służy do porównywania kolejnych bitów dwóch ciągów bitów, z możliwością maskowania wybranych bitów. Długość ciągu słów 16-bitowych (dla funkcji MSKCMPW) lub 32-bitowych (dla funkcji MSKCMPD) określa parametr LEN.

Po doprowadzeniu sygnału wejściowego rozpoczynane jest porównywanie bitów pierwszego zdefiniowanego ciągu bitów I1 z odpowiednimi bitami drugiego ciągu I2. Porównywanie trwa do czasu, aż zostanie wykryta różnica lub porównane zostaną ostatnie bity.

Parametr wejściowy BIT wykorzystywany jest do zapamiętywania numeru bitu, od którego należy rozpocząć następane szukanie (0 oznacza pierwszy bit ciągu). Parametr wyjściowy BN wykorzystywany jest do zapamiętywania numeru ostatnio porównywanego bitu (1 oznacza pierwszy bit ciągu). Jeśli BN i BIT zostanie przypisany ten sam adres, kolejna operacja porównywania rozpocznie się na kolejnej pozycji po wykrytej niezgodności stanu bitów.

Jeżeli następane szukanie ma być rozpoczęte w innym miejscu, można wprowadzić inne wartości dla BIT i BN. Jeżeli wartość parametru BIT jest większa od długości ciągu, przed rozpoczęciem następnego porównywania wartość tego parametru ustawiana jest na 0.

Wszystkie bity w parametrów I1 i I2 są takie same

Jeśli wszystkie bity w obydwu porównywanych ciągach są identyczne, wyjście MC jest ustawiane na 0, a wyjście BN na najwyższy numer bitu w porównywanych ciągach. Następnie porównywanie zostaje przerwane. Następne wywołanie MSKCMPW spowoduje jej wyzerowanie.

Znalezienie bitów o różnych wartościach

Gdy dwa porównywane bity różnią się, funkcja sprawdza stan odpowiedniego bitu w ciągu bitów M (maska). Jeśli bit maski posiada wartość 1, porównywanie jest kontynuowane do czasu znalezienia następnych dwóch różniących się bitów lub do końca słowa bitowego.

Jeśli bit maski posiada wartość 0, blok funkcyjny wykonuje następujące operacje:

1. Ustawia stan odpowiedniego bitu maski M na 1.
2. Ustawia wyjście MC na 1 (wykrycie niezgodności).
3. Aktualizuje ciąg bitów Q tak, aby był zgodny ze zaktualizowaną maską.
4. Ustawia wartość parametru BN na numer bitu, dla którego wykryto niezgodność.
5. Przerzywa porównywanie.

(enable)	-	MASK	-	
		COMP		
		WORD		
(parametr wejściowy I1)	-	I1 MC	-	(wykrycie niezgodności)
		LEN		
		00001		
(parametr wejściowy I2)	-	I2 Q	-	(parametr wyjściowy Q)
(maska)	-	M BN	-	(Numer ostatniego porównywanego bitu)
(numer bitu)	-	BIT		

Parametry

Parametr	Opis
enable	Sygnal wejściowy. Po doprowadzeniu sygnału wejściowego wykonywana jest operacja porównania z maskowaniem.
I1	Adres pierwszego ciągu porównanych bitów.
I2	Adres drugiego ciągu porównanych bitów.
M	Adres maski.
BIT	Adres zmiennej (rejestru), w którym przechowywany jest numer bitu, od którego powinna się zacząć kolejna operacja porównywania.
MC	Wyjście informujące o wykryciu bitów o różnych wartościach.
Q	Kopia maski.
BN	Numer ostatniego porównywanego bitu.
LEN	Liczba słów bitowych ciągu słów, z którego wybierany jest bit, którego wartość ma zostać zmieniona (maksymalnie 256).

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
I1		o	o	o	o	o	o	•	•	•		
I2		o	o	o	o	o	o	•	•	•		
M		o	o	o	o	o†	o	•	•	•		
BIT		•	•	•	•	•	•	•	•	•	•	
LEN											•‡	
MC	•											•
Q		o	o	o	o	o†	o	•	•	•		
BN		•	•	•	•	•	•	•	•	•		

- Typ może być wykorzystany jako dany parametr bloku funkcyjnego.
- o Dopuszczalne jest stosowanie wyłącznie danych typu WORD, nie można stosować zmiennych typu DWORD.
- † Można stosować wyłącznie zmienne systemowe typu %SA, %SB i %SC. Nie można korzystać ze zmiennych typu %S.
- ‡ Maksymalna wartość stałej może wynosić 4095 dla danych typu WORD i 2047 dla danych typu DWORD.

Przykład

W zamieszczonym poniżej przykładzie blok funkcyjny MSKCOMPW wykonywany jest w pierwszym cyklu pracy sterownika. Zmienne %M0001 do %M0016 są porównywane ze zmiennymi %M0017 do %M0032. Maska określana jest poprzez zmienne %M0033 do %M0048. Wartość podana poprzez %R0001 określa numer bitu, od którego ma być rozpoczęte porównywanie dwóch ciągów wejściowych. Przed rozpoczęciem wykonywania tego bloku funkcyjnego, wymienione powyżej zmienne mają następujące wartości:

(I1) –%M0001 = 6C6Ch =

0	1	1	0	1	1	0	0	0	1	1	0	1	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(I2) –%M0017 = 6D6Fh =

0	1	1	0	1	1	0	1	0	1	1	0	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(M/Q) –%M0033 = 000Fh =

0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(BIT/BN) –%R0001 = 0

(MC) –%Q0001 = OFF

Wartości tych zmiennych po wykonaniu bloku funkcyjnego są następujące:

(I1) –%M0001 =

0	1	1	0	1	1	0	0	0	1	1	0	1	1	0	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(I2) –%M0017 =

0	1	1	0	1	1	0	1	0	1	1	0	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(M/Q) – %M0033 =

0	0	0	0	0	0	0	1	0	0	0	0	1	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

(BIT/BN) –%R0001 = 9

(MC) –%Q0001 = ON

Szczelbel programu sterującego

FST_SCN	MASK	
%M0001 -	COMP-WORD	%Q0001 (S) -
%M0017 -	I1 MC LEN 00001	
%M0033 -	I2 Q - %M0033	
%R0001 -	M BN - %R0001	
	BIT	

W przykładzie pokazanym powyżej należy zwrócić uwagę na zastosowanie styku FST_SCN do wymuszenia wykonywania tego bloku wyłącznie jeden raz. W przypadku braku takiego styku, wykonywanie tego bloku byłoby powtarzane, a wynik byłby obliczany bez potrzeby.

Rozdział 9

Funkcje do przesyłania danych

Funkcje z tej grupy umożliwiają przemieszczanie danych. W rozdziale tym opisano następujące funkcje do przesyłania danych:

Oznaczenie skrótowe	Element	Opis	Strona
MOVE	Przemieszczenie	Kopiowanie danych, traktowanych jako pojedyncze bity. Maksymalna, dopuszczalna długość wynosi 256 słów, za wyjątkiem Move_BIT, dla której wielkość ta wynosi 256 bitów. Można przemieszczać dane różnych typów, bez konieczności ich uprzedniej konwersji.	9-2
BLKMOV	Przemieszczenie bloku	Kopiowanie grupy siedmiu stałych wartości do określonego obszaru pamięci sterownika. Stałe te są podawane jako parametru wejściowe.	9-5
BLKCLR	Zerowanie fragmentu pamięci	Zerowanie określonego bloku pamięci sterownika. Funkcja ta może zostać wykorzystana do zerowania pamięci zmiennych dyskretnych (%I, %Q, %M, %G, or %T) i pamięci zmiennych rejestrowych (%R, %AI i %AQ). Maksymalna, dopuszczalna długość wynosi 256 słów.	9-7
SHFR	Rejestr przemieszczający	Wstawienie jednego lub więcej słów w określone miejsce pamięci. Maksymalna, dopuszczalna długość wynosi 256 słów.	9-8
BITSEQ	Przemieszczenie jedynek	Przemieszczenie sekwencji bitów w ciągu bitów. Maksymalna, dopuszczalna długość wynosi 256 bitów.	9-11
COMMREQ	Inicjalizacja komunikacji z jednym z modułów sterownika	Funkcja ta pozwala na nawiązanie komunikacji sterownika z jednym z wyspecjalizowanych modułów dodatkowych, jak np. z modułem komunikacyjnym Genius lub z modułem programowalnego koprocesora.	9-14

MOVE (BIT, INT, WORD, REAL)

Funkcja MOVE służy do kopiowania danych (traktowanych jako pojedyncze bity) z określonego miejsca pamięci w inne. Ponieważ dane są przesyłane jako bity, nowy adres nie musi być tego samego typu co adres oryginalny.

Funkcja MOVE posiada dwa parametry wejściowe i dwa parametry wyjściowe. Po doprowadzeniu sygnału wejściowego, dane są kopiowane bit po bicie z miejsca pamięci określonego przez parametr IN w miejsce określone przez parametr Q. Jeśli dane przesyłane są pomiędzy obszarami przypisanymi zmiennym dyskretnym (np. z obszaru pamięci przypisanego zmiennym typu %I do obszaru przypisanego zmiennym %T), bity towarzyszące, sterujące wymuszoną zmianą wartości tych zmiennych, są również przesyłane. Dane oryginalne nie ulegają zmianie, chyba że zakresy adresów parametrów wejściowego i wyjściowego pokrywają się.

W przypadku danych typu BIT należy uwzględnić jeszcze pewien dodatkowy aspekt. Jeżeli ciąg danych typu BIT określony przy pomocy parametru Q nie zawiera wszystkich bitów bajtu, bity towarzyszące powiązane z tym bajtem (nie mieszczące się w tym ciągu) są ustawiane na wartość 0 po doprowadzeniu sygnału do funkcji MOVE_BIT.

Parametr wejściowy IN może być adresem zmiennej lub stałą. Jeżeli parametrem jest wartość stała, w miejscu określonym przez parametr wyjściowy Q zapisywana jest wartość stała. Jeśli przykładowo parametrem wejściowym IN jest stała 4, w pamięci pod adresem Q zapisywana jest wartość 4. Jeśli parametr ?? ma wartość większą od 1 a parametr wejściowy IN jest stałą, to zostaje ona skopiowana do odpowiedniej liczby komórek pamięci, określonej parametrem LEN, począwszy od miejsca adresowanego przez parametr Q. Przykładowo jeżeli parametrem wejściowym jest stała 9, długość wynosi 4, w pamięci, pod adresem wskazywanym przez Q oraz w trzech dalszych pozycjach zostanie zapisana wartość 9.

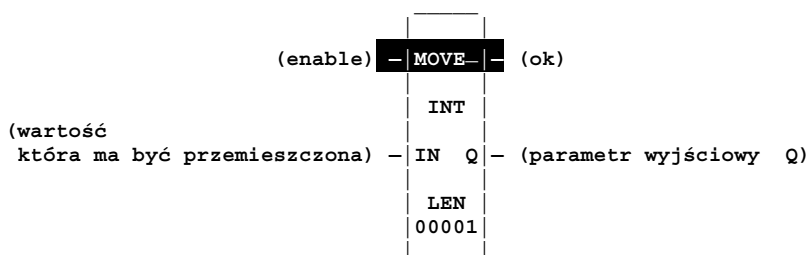
Parametr LEN określa liczbę:

- Słów, które mają być przemieszczone za pomocą funkcji MOVE_INT i MOVE_WORD.
- Bitów, które mają być przemieszczone za pomocą funkcji MOVE_BIT.
- Liczb rzeczywistych, które mają być przemieszczone za pomocą funkcji MOVE_REAL.

Uwaga

Operacje na liczbach rzeczywistych można realizować wyłącznie w jednostkach centralnych 35x i 36x, oprogramowanie systemowe wer. 9 lub nowsza oraz we wszystkich wersjach jednostki centralnej CPU352.

Sygnał wyjściowy jest wysyłany zawsze, po doprowadzeniu sygnału wejściowego.



Parametry

Parametr	Opis
enable	Sygnal wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
IN	Wartość stała lub adres zmiennej, która ma zostać przemieszczona. W przypadku funkcji MOVE_BIT, można podać dowolną zmienną dyskretną, bez względu na ilość zajmowanych bitów. Nie mniej jednak, w czasie pracy wyświetlanych jest 16 bitów, począwszy od zadanego adresu.
ok	Sygnal wyjściowy wysyłany po doprowadzeniu sygnału wejściowego.
Q	Po dokonaniu przemieszczenia, wartość określona przez parametr IN jest kopiowana w miejsce określone przez parametr Q. Nie mniej jednak, w czasie pracy wyświetlanych jest 16 bitów, począwszy od zadanego adresu.
LEN	Liczba słów bitowych lub bitów, które mają zostać przemieszczone. Dla funkcji MOVE_INT oraz MOVE_WORD parametr LEN musi zawierać się pomiędzy 1 a 256. Dla funkcji MOVE_BIT, gdy parametr IN określa wartość stałą, parametr LEN musi zawierać się pomiędzy 1 a 16, w przeciwnym wypadku, parametr LEN musi przyjmować wartość z zakresu 1 do 256.

Uwaga

W jednostkach centralnych 351, 352 i jednostkach centralnych 36x, funkcje MOVE_INT i MOVE_WORD nie dopuszczają możliwości nakładania się adresów IN i Q, jeżeli adres IN jest mniejszy od adresu Q. Przykładowo, w przypadku wartości: IN=%R0001, Q=%R0004, LEN=5 (słów), zawartości wskazywane przez %R0007 i %R0008 będą nieokreślone; nie mniej jednak, dla wartości: Q=%R0001, IN=%R0004, LEN=5 (słów) otrzymany zostanie poprawny wynik.

Należy również zwrócić uwagę, że możliwość wykonywania działań na liczbach rzeczywistych, a więc możliwość korzystania z funkcji MOVE_REAL, jest obecnie dostępna wyłącznie w jednostkach centralnych serii 35x i 36x (oprogramowanie systemowe ver. 9 i nowsze oraz we wszystkich wersjach jednostki centralnej CPU352).

Dopuszczalne typy parametrów

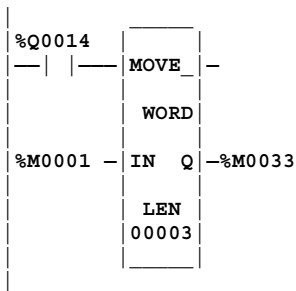
Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN		•	•	•	•	o	•	•	•	•	•	
ok	•											•
Q		•	•	•	•	o†	•	•	•	•		

Uwaga: W przypadku danych REAL, dopuszczalne jest stosowanie wyłącznie typu %R, %AI i %AQ.

- Dopuszczalny typ parametru dla danych typu BIT, INT i WORD. W funkcji MOVE_BIT, wartości zmiennych dyskretnych %I, %Q, %M i %T nie muszą być podawane z uwzględnieniem podziału na bajty.
- o Dopuszczalny typ parametru do operacji na danych BIT lub Word. Nie można go wykorzystywać w przypadku operacji na danych typu INT.
- † Można stosować wyłącznie zmienne systemowe typu %SA, %SB i %SC. Nie można korzystać ze zmiennych typu %S.

Przykład 1

Po ustawieniu wartości zmiennej %Q0014 na 1, z pamięci o adresie %M0001 skopiowanych zostanie 48 bitów, do pamięci o adresie %M0033. Pomimo, że obszar pamięci do którego wykonywane jest kopiowanie pokrywa się z 16 bitami wejściowymi, operacja wykonywana jest poprawnie (za wyjątkiem jednostek centralnych 351 i 352, zgodnie z przytoczoną wcześniej uwagą).



Przed wykonaniem funkcji Move:

INPUT (%M0001 do %M0048)

1

%M0016	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0
%M0032	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
%M0048	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Po wykonaniu funkcji Move:

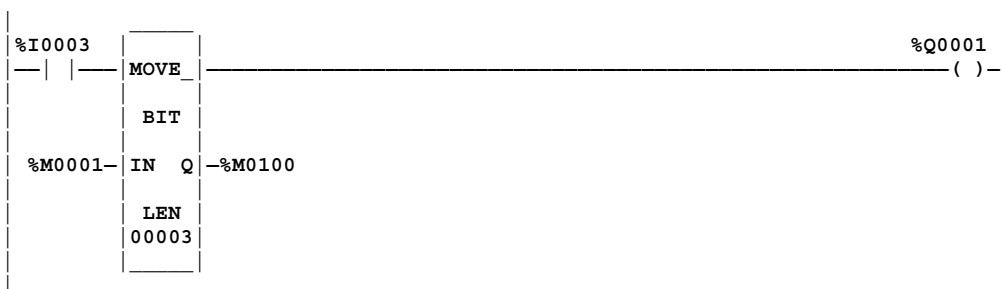
INPUT (%M0033 do %M0080)

33

%M0048	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0
%M0064	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
%M0080	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Przykład 2

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0001 na 1, powoduje przemieszczenie trzech bitów %M0001, %M0002 i %M0003 do odpowiednio %M0100, %M0101 i %M0102. Zmienna przekaźnika %Q0001 jest ustawiana na 1.



BLKMOV (INT, WORD, REAL)

Funkcja BLKMOV służy do kopiowania grupy siedmiu stałych wartości do określonego obszaru pamięci sterownika.

Uwaga

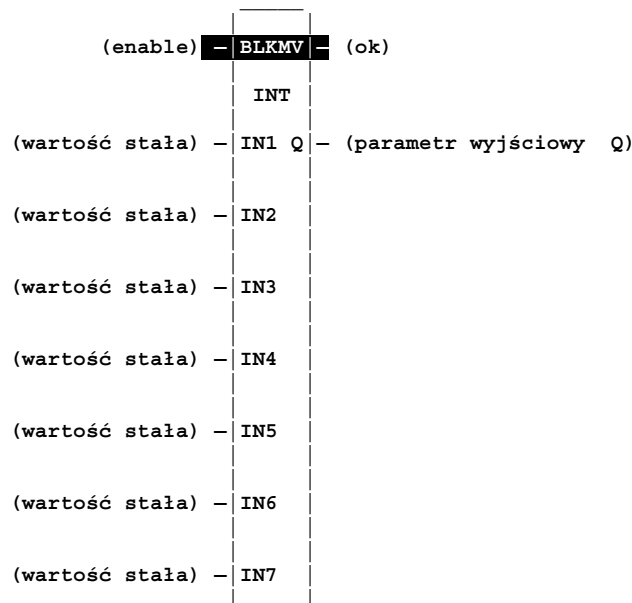
Operacje na liczbach rzeczywistych można realizować wyłącznie w jednostkach centralnych 35x i 36x, oprogramowanie systemowe wer. 9 lub nowsza oraz we wszystkich wersjach jednostki centralnej CPU352.

Funkcja BLKMOV posiada osiem parametrów wejściowych i dwa parametry wyjściowe. Po doprowadzeniu sygnału wejściowego wartości stałe są kopiowane w miejsce zaadresowane przez parametr Q (tzn. pierwsza wartość IN1 jest kopiowana w miejsce określone przez Q, a następné sześć wartości w kolejne miejsca pamięci). Parametr Q nie może być parametrem wejściowym innego bloku funkcyjnego.

Uwaga

Dla funkcji MOVE_INT, wartości parametrów IN1 - IN7 wyświetlane są jako liczby dziesiętne ze znakiem. Dla funkcji BLKMOV_WORD parametry te są wyświetlane jako liczby heksadecymalne. Dla funkcji BLKMOV_REAL parametry IN1-IN7 wyświetlane są jako liczby rzeczywiste.

Sygnal wyjściowy jest wysyłany zawsze, po doprowadzeniu sygnału wejściowego.



Parametry

Parametr	Opis
enable	Sygnal wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
IN1—IN7	Wartości stałe, które mają zostać skopiowane.
ok	Sygnal wyjściowy wysyłany po doprowadzeniu sygnału wejściowego.
Q	Miejsce w pamięci, w którym ma być umieszczona pierwsza z kopiowanych wartości. Do miejsca określonego przez parametr Q kopiowana jest wartość IN1.

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN1—IN7											•	
ok	•											•
Q		•	•	•	•	o†	•	•	•	•		

Uwaga: W przypadku danych REAL, dopuszczalne jest stosowanie wyłącznie danych %R, %AI i %AQ.

- Typ może być wykorzystany jako dany parametr bloku funkcyjnego.
- o Typ może być wykorzystany wyłącznie jako parametr do operacji na danych typu Word. Nie można go wykorzystywać w przypadku operacji na danych typu INT i REAL.
- † Można stosować wyłącznie zmienne systemowe typu %SA, %SB i %SC. Nie można korzystać ze zmiennych typu %S.

Uwaga

Możliwość wykonywania operacji na liczbach rzeczywistych dostępna jest wyłącznie w jednostkach centralnych 35x i 36x, oprogramowanie systemowe wer. 9 lub nowsza oraz we wszystkich wersjach jednostki centralnej CPU352. Tylko te jednostki centralne obsługują funkcję BLKMOV_REAL.

Przykład

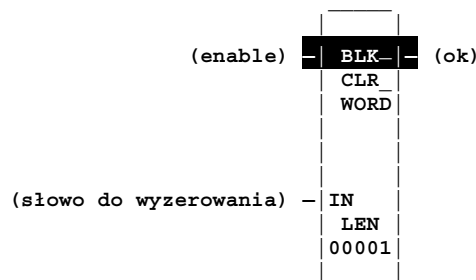
W zamieszczonym poniżej przykładzie, po zmianie wartości zmiennej o nazwie pomocniczej FST_SCN na 1, funkcja BLKMOV kopiuje siedem wartości stałych do obszaru pamięci, określonego przez adresy %R0010 do %R0016.

FST_SCN	BLKMOV	INT
CONST +32767	IN1 Q	%R0010
CONST -32768	IN2	
CONST +00001	IN3	
CONST +00002	IN4	
CONST -00002	IN5	
CONST -00001	IN6	
CONST +00001	IN7	

BLKCLR (WORD)

Funkcja BLKCLR służy do wyzerowania określonego bloku pamięci sterownika.

Funkcja BLKCLR posiada dwa parametry wejściowe i jeden parametr wyjściowy. Po doprowadzeniu sygnału wejściowego, wszystkie bity obszaru pamięci o długości określonej przez LEN i adresie początkowym zadanym parametrem IN, zostają wyzerowane. Jeśli ma zostać wyzerowany obszar pamięci przypisany zmiennym dyskretnym (typu %I, %Q, %M, %G lub %T), bity towarzyszące, sterujące wymuszoną zmianą wartości tych zmiennych, zostają również wyzerowane. Funkcja przesyła sygnał wyjściowy.



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału do bloku funkcyjnego powoduje wyzerowanie zadanego obszaru pamięci.
IN	Adres pierwszego słowa pamięci, które ma zostać wyzerowane.
ok	Sygnał wyjściowy wysyłany po doprowadzeniu sygnału wejściowego.
LEN	Parametr LEN musi mieć wartość z zakresu 1 do 256 słów.

Dopuszczalne typy parametrów

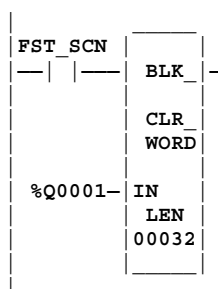
Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN		•	•	•	•	•†	•	•	•	•		
ok	•											•

• Dozwolony typ parametru.

† Można stosować wyłącznie zmienne systemowe typu %SA, %SB i %SC. Nie można korzystać ze zmiennych typu %S.

Przykład

W zamieszczonym poniżej przykładzie podczas rozruchu sterownika zerowane są 32 słowa pamięci typu %Q, począwszy od adresu %Q0001.



SHFR (BIT, WORD)

Blok funkcyjny SHFR służy do wstawienia jednego bitu (lub słowa bitowego) w określone miejsce w pamięci sterownika, przesunięcie zawartości obszaru pamięci rozpoczynającego się w tym miejscu (posiadającego określoną długość) o jeden bit (lub słowo bitowe) w lewo oraz przemieszczenie bitu (lub słowa bitowego), "wypchniętego" z ostatniego miejsca obszaru pamięci, w inne miejsce. Przykładowo, można wstawić jedno słowo do obszaru pamięci o długości pięciu słów. W wyniku takiej operacji, jedno słowo danych zostanie "wypchnięte" poza ten obszar pamięci.

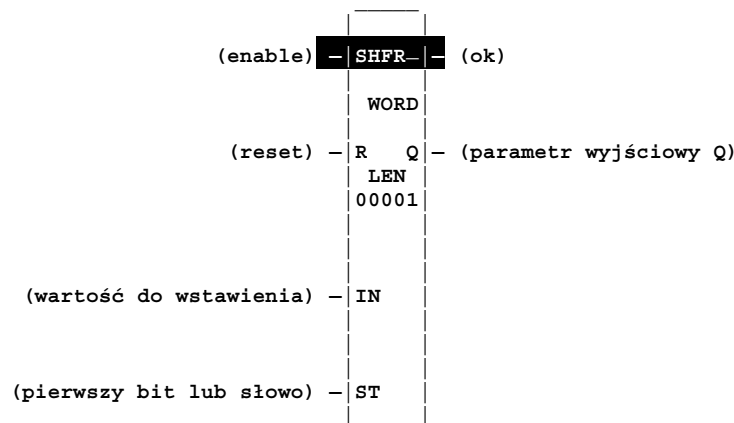
Uwaga

Pokrywanie się adresów parametrów wejściowych i wyjściowych w przypadku funkcji operujących na ciągach wielu słów bitowych mogą być przyczyną uzyskania nieoczekiwanych wyników.

Funkcja SHFR posiada cztery parametry wejściowe i dwa parametry wyjściowe. Wejście zerujące R powoduje wyzerowanie wszystkich bitów lub słów w przemieszczanym obszarze pamięci i ma wyższy priorytet w stosunku do wejścia enable. Jeżeli aktywny jest sygnał Reset obszaru pamięci, którego początek zawiera parametr ST (adres pierwszego bitu lub słowa bitowego rejestru) o długości LEN (długość rejestru w bitach lub słowach bitowych) jest wypełniany zerami.

Po doprowadzeniu do bloku funkcyjnego sygnału wejściowego, jeśli równocześnie na wejście reset nie jest podawany sygnał, każdy bit lub słowo bitowe rejestru przemieszczane są w lewo. Najbardziej znaczący element rejestru IN przemieszczany jest w wolne miejsce, którego adres podaje parametr ST. Zawartość przemieszczanego bloku pamięci jest dostępna dla innych bloków funkcyjnych programu sterującego (bity tego bloku pamięci mogą być wykorzystywane przez inne bloki funkcyjne).

Sygnał wyjściowy jest wysyłany zawsze, po doprowadzeniu sygnału wejściowego.



Parametry

Parametr	Opis
enable	Jeżeli do bloku funkcyjnego doprowadzany jest sygnał i jednocześnie nie jest doprowadzany sygnał na wejście R, rejestr przemieszczający wykonuje operację przemieszczenia.
R	Jeżeli na wejście R doprowadzany jest sygnał, rejestr wskazywany przez ST wypełniany jest zerami.
IN	IN określa wartość, która ma zostać skopiowana do pierwszego bitu lub słowa bitowego rejestru przemieszczającego. Dla funkcji SHFR_BIT można tu wykorzystać dowolną zmienną dyskretną, bez uwzględniania podziału na bajty. Nie mniej jednak, w czasie pracy wyświetlanych jest 16 bitów, począwszy od zadanego adresu.
ST	ST określa adres pierwszego bitu lub słowa bitowego rejestru przemieszczającego. Dla funkcji SHFR_BIT można tu wykorzystać dowolną zmienną dyskretną, bez uwzględniania podziału na bajty. Nie mniej jednak, w czasie pracy wyświetlanych jest 16 bitów, począwszy od zadanego adresu.
ok	Sygnał wyjściowy wysyłany po doprowadzeniu sygnału wejściowego.
Q	Adres, pod który ma zostać skopiowany bit lub słowo bitowe "wypchnięte" z rejestru przemieszczającego po wykonaniu operacji. Dla funkcji SHFR_BIT można tu wykorzystać dowolną zmienną dyskretną, bez uwzględniania podziału na bajty. Nie mniej jednak, w czasie pracy wyświetlanych jest 16 bitów, począwszy od zadanego adresu.
LEN	Ilość przemieszczanych rejestrów. Dla funkcji SHFR_WORD, parametr LEN musi mieć wartość z zakresu 1 do 256 słów. Dla funkcji SHFR_BIT, parametr LEN musi mieć wartość z zakresu 1 do 256 bitów.

Dopuszczalne typy parametrów

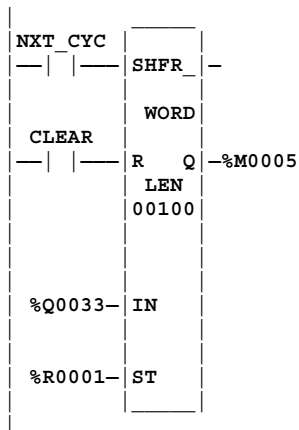
Parametr	Sygnał	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
R	•											
IN		•	•	•	•	•	•	•	•	•	•	
ST		•	•	•	•	•†	•	•	•	•		
ok	•											•
Q		•	•	•	•	•†	•	•	•	•		

- Dla danych typu BIT i WORD, zmienne tego typu mogą być wykorzystywane jako parametry bloku funkcyjnego. W funkcji SHFR_BIT, wartości zmiennych dyskretnych %I, %Q, %M i %T nie muszą być podawane z uwzględnieniem podziału na bajty.
- † Można stosować wyłącznie zmienne systemowe typu %SA, %SB i %SC. Nie można korzystać ze zmiennych typu %S.

Przykład 1

W zamieszczonym poniżej przykładzie, rejestr przemieszczający wykorzystuje pamięć od %R0001 do %R0100. Jeżeli aktywny jest sygnał zerowania CLEAR, słowa rejestru przemieszczającego są zerowane.

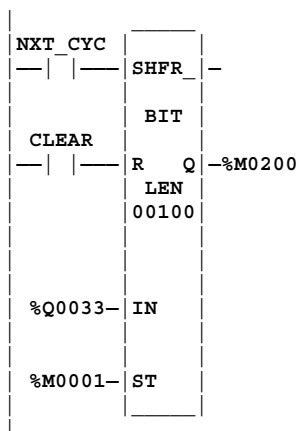
Jeżeli wartość zmiennej NXT_CYC jest równa 1, a sygnał CLEAR nie jest doprowadzany, słowo z obszaru adresowanego przez %Q0033 jest przemieszczane do rejestru przemieszczającego o adresie %R0001. Słowo "wypchnięte" z rejestru przemieszczającego, o adresie %R0100, jest zapisywane w obszarze o adresie %M0005.



Przykład 2

W zamieszczonym poniżej przykładzie, rejestr przemieszczający wykorzystuje pamięć od %R0001 do %R0100. Jeżeli aktywny jest sygnał zerowania CLEAR, funkcja SHFR wypełnia zerami obszar pamięci ograniczony adresami %M0001 do %M0100.

Jeżeli wartość zmiennej NXT_CYC jest równa 1 oraz nie jest doprowadzany sygnał CLEAR, funkcja SHFR przemieszcza dane bit po bicie, z %M0001 do %M0100. Bit %Q0033 jest zapisywany do bitu %M0001, natomiast bit "wypchnięty" jest zapisywany do %M0200.



BITSEQ (BIT)

Blok funkcyjny BITSEQ wykorzystywany jest do cyklicznego przemieszczania w lewo lub w prawo bitu o wartości 1 w pewnym obszarze pamięci, w którym pozostałe bity mają wartość 0. Blok funkcyjny BITSEQ posiada pięć parametrów wejściowych i jeden parametr wyjściowy. Wynik działania tego bloku funkcyjnego zależy od poprzedniej wartości parametru EN, zgodnie z poniżej zamieszczoną tabelą.

Sygnal na wejściu R w bieżącym cyklu	Sygnal na wejściu EN w poprzednim cyklu	Sygnal na wejściu EN w bieżącym cyklu	Działanie bloku funkcyjnego BIT_SEQ
OFF	OFF	OFF	Brak działania.
OFF	OFF	ON	Przesunięcie bitu o wartości równej 1 o jedno miejsce w lewo lub w prawo.
OFF	ON	OFF	Brak działania.
OFF	ON	ON	Brak działania.
ON	ON/OFF	ON/OFF	Ustawienie warunków początkowych.

Podanie sygnału na wejście zerujące R ma większy priorytet od podania sygnału na wejście EN i zawsze powoduje przejście do stanu początkowego bloku. Jeżeli doprowadzany jest sygnał na wejście zerujące R, bieżący numer bitu jest przepisywany z parametru wejściowego STEP. Jeżeli parametr STEP nie jest wykorzystywany, wartość 1 przypisywana jest pierwszemu bitowi obszaru pamięci, na którym działa blok funkcyjny BITSEQ. Wartości wszystkich pozostałych bitów tego obszaru są ustawiane na 0.

Gdy na wejście "enable" podany zostanie sygnał wyjściowy i jednocześnie nie jest on podawany na wejście zerujące R, bit posiadający do tej pory wartość 1 zostaje wyzerowany. Numer bieżącego bitu jest inkrementowany lub dekrementowany, w zależności od wartości parametru DIR. Wartość bitu o zmienionym numerze zostaje ustawiona na 1.

- Jeśli podczas wykonywania bloku funkcyjnego bieżący numer bitu o wartości równej 1 jest zwiększany i przekroczy dopuszczalny zakres ($1 \leq \text{bieżący numer} \leq \text{LEN}$), numer ten jest ustawiany na 1.
- Jeśli podczas wykonywania bloku funkcyjnego bieżący numer bitu o wartości równej 1 jest zmniejszany i przekroczy dopuszczalny zakres ($1 \leq \text{bieżący numer} \leq \text{LEN}$), numer ten jest ustawiany na LEN.

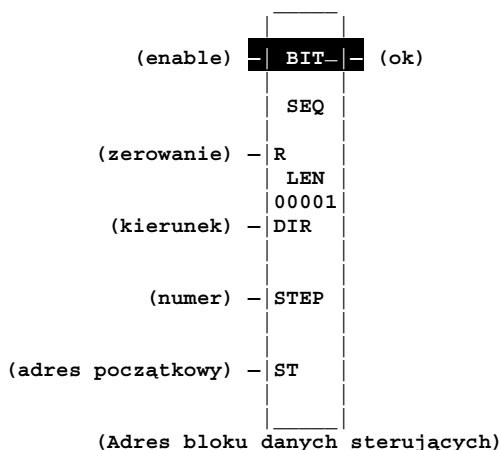
Parametr ST jest parametrem opcjonalnym. Jeżeli nie jest wykorzystywany, blok funkcyjny BITSEQ pracuje zgodnie z przedstawionym powyżej opisem, żadne bity nie są jednak ustawiane czy zerowane. W zasadzie, blok funkcyjny BITSEQ w takim przypadku powoduje cykliczne przemieszczanie bieżącego numeru bitu w dopuszczalnym zakresie.

Blok danych sterujących

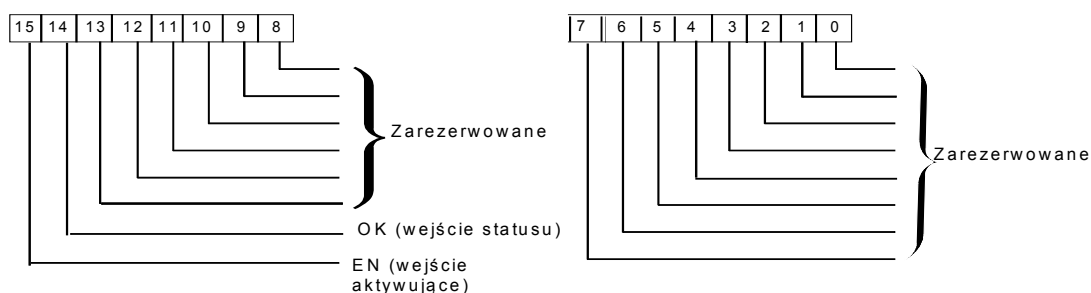
Każdy blok funkcyjny BITSEQ wykorzystuje trzy słowa (rejstry) pamięci typu %R, do zapamiętania następujących parametrów:

numer bieżącego bitu	słowo 1
długość obszaru pamięci (w bitach)	słowo 2
słowo sterujące	słowo 3

W bloku funkcyjnym BITSEQ należy podać adres początkowy dla tych trzech słów (rejestrów), bezpośrednio pod symbolem graficznym (proszę porównać z zamieszczonym w dalszej części przykładem).



Słowo sterujące zapamiętuje stan wejść i wyjść cyfrowych, powiązanych z danym blokiem funkcyjnym, zgodnie z przedstawionym poniżej formatem:



Uwaga

Bity 0 do 13 nie są wykorzystywane. Należy również zwrócić uwagę, że w parametrze STEP bity numerowane są od 1 do 16, a **NIE** 0 do 15.

Parametry

Parametr	Opis
adres	Adres pierwszego z trzech rejestrów, w których przechowywany jest bieżący numer bitu o wartości równej 1, długość wykorzystywanego obszaru pamięci oraz inne informacje, sterujące przebiegiem wykonywania tego bloku funkcyjnego.
enable	Jeżeli do bloku funkcyjnego doprowadzany jest sygnał, nie był on doprowadzany w poprzednim cyklu, a jednocześnie sygnał nie jest doprowadzany do wejścia R, blok funkcyjny BITSEQ wykonuje przemieszczenie bitu o wartości równej 1 w górę lub w dół.
R	Podanie sygnału na wejście R powoduje ustawienie warunków początkowych, tzn. skopiowanie bieżącego numeru bitu o wartości 1 z parametru STEP (lub ustawienie tego numeru na 1 w przypadku braku parametru STEP), wyzerowanie obszaru pamięci przypisanego do bloku funkcyjnego BITSEQ oraz ustawienie wartości bitu o numerze STEP na 1.
DIR	DIR określa kierunek przemieszczania bitu o wartości równej 1. Jeśli na wejście DIR podawany jest sygnał, bit przemieszczany jest w górę (bieżący numer rośnie). Jeśli sygnał nie jest podawany, bit przemieszczany jest w dół (bieżący numer maleje).
STEP	STEP jest numerem początkowym bitu o wartości równej 1.
ST	Adres pierwszego bitu obszaru pamięci, wykorzystywanego przez blok funkcyjny BITSEQ.
ok	Sygnał wyjściowy wysyłany po doprowadzeniu sygnału wejściowego.
LEN	Parametr LEN musi mieć wartość z zakresu 1 do 256 bitów.

Uwaga

W czasie sprawdzania wielokrotnego wykorzystania zmiennych w przekaźnikach, dla bloku funkcyjnego BITSEQ sprawdzanych jest 16 bitów parametru ST, nawet jeżeli parametr LEN ma wartość mniejszą od 16.

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
adres								•				
enable	•											
R	•											
DIR	•											
STEP		•	•	•	•		•	•	•	•	•	•
ST		•	•	•	•	•†	•	•	•	•		•
ok	•											•

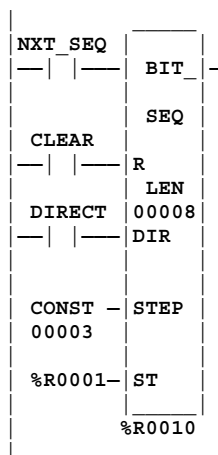
• Dozwolony typ parametru.

† Można stosować wyłącznie zmienne systemowe typu %SA, %SB i %SC. Nie można korzystać ze zmiennych typu %S.

Przykład

W zamieszczonym poniżej przykładzie, blok funkcyjny BITSEQ wykorzystuje pamięć o adresie %R0001. Blok danych sterujących zajmuje rejestry %R0010, %R0011 i %R0012. Po doprowadzeniu sygnału wejściowego CLEAR następuje wyzerowanie obszaru pamięci przypisanego do bloku funkcyjnego oraz ustawienie 3 bitu jako bitu bieżącego. Pierwszych 8 bitów rejestru %R0001 ustawianych jest na 0.

Jeżeli doprowadzony zostanie sygnał NXT_SEQ i nie jest doprowadzany sygnał CLEAR, następuje ustawienie wartości bitu 3 na 0 oraz ustawienie wartości 1 dla bitu 2 lub 4 (w zależności od tego, czy doprowadzany jest sygnał wejściowy DIR).



COMMREQ

Blok funkcyjny COMMREQ służy do komunikacji z modułami wyspecjalizowanymi, jak np. z modułem komunikacyjnym GENIUS lub z modułem programowalnego koprocesora.

Uwaga

Informacje podane w tym punkcie mają na celu zapoznanie użytkownika z ogólnym formatem bloku funkcyjnego COMMREQ. Do korzystania z bloków funkcyjnych COMMREQ potrzebne są dodatkowe informacje. Są one zawarte w dokumentacji technicznej modułów.

Blok funkcyjny COMMREQ posiada trzy parametry wejściowe i jeden parametr wyjściowy. Po doprowadzeniu do bloku funkcyjnego sygnału wejściowego, do modułu wysyłany jest blok danych sterujących. Początek tego bloku określony jest za pomocą parametru IN. Moduł, z którym ma zostać nawiązana komunikacja, jest identyfikowany poprzez parametr SYSID, który zawiera numer kasety i gniazda, w którym dany moduł jest zainstalowany.

Po przesłaniu danych do wyszczególnionego modułu, blok funkcyjny COMMREQ może czekać na odpowiedź lub może kontynuować wykonywanie programu sterującego. Jeżeli w bloku sterującym określono, że program ma nie czekać na odpowiedź, zawartość bloku danych sterujących jest przesyłana do modułu docelowego, a jednostka centralna sterownika kontynuuje wykonywanie programu sterującego bez żadnych przerw. (Ignorowanie parametru określającego dopuszczalny czas oczekiwania). Jest to domyślny tryb **NOWAIT**.

Jeżeli w bloku sterującym określono, że program ma czekać na odpowiedź, zawartość bloku danych sterujących jest przesyłana do modułu docelowego, a jednostka centralna sterownika oczekuje na odpowiedź modułu. Maksymalny czas oczekiwania jednostki centralnej na odpowiedź modułu jest określony w bloku danych sterujących. Jeśli czas ten upłynie, a moduł nie odpowiada, wykonywanie programu sterującego jest kontynuowane. Jest to tryb **WAIT**.

Blok funkcyjny COMMREQ ustawia wartość parametru wyjściowego FT (błędne wykonanie) na 1, jeśli zachodzi jeden z następujących przypadków:

1. Pod wyszczególnionym adresem nie ma żadnego modułu.
2. Zadanie zlecane wyszczególnionemu modułowi nie może być przez ten moduł wykonane.
3. Długość przesłanego bloku danych wynosi 0.
4. Zły adres pamięci, w której przechowywany jest status urządzenia (część bloku danych sterujących). Może to być spowodowane wybraniem zakresu pamięci przyporządkowanego innemu typowi zmiennych lub adresu przekraczającego zakres dozwolony dla zmiennych danego typu.

Blok danych sterujących

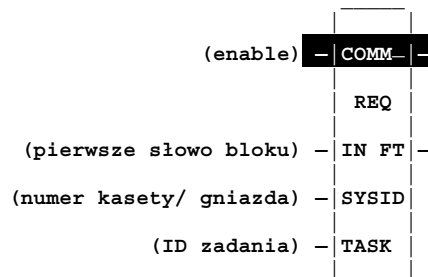
Blok danych sterujących zawiera dodatkowe informacje wymagane przez blok funkcyjny COMMREQ.

Adres początkowy tego bloku zawarty jest w parametrze IN bloku funkcyjnego. Może on wskazywać dowolny obszar pamięci zorientowanej rejestrowo (%R, %AI lub %AQ). Długość bloku danych sterujących zależy od ilości danych wysyłanych do wyspecjalizowanego modułu dodatkowego.

Blok ten posiada następującą strukturę:

Długość (w słowach)	adres
Tryb pracy (WAIT lub NOWAIT)	adres + 1
Pamięć wskaźnika statusu	adres + 2
Adres względny wskaźnika statusu	adres + 3
Maksymalny czas oczekiwania na odpowiedź	adres + 4
Maksymalny czas przeznaczony na komunikację	adres + 5
Blok danych	adres + 6
	do
	adres + 133

Informacje, które mają być zawarte w bloku danych sterujących można przesłać do odpowiednich obszarów pamięci sterownika za pomocą opisywanych wcześniej bloków funkcyjnych (np. MOVE lub BLKMOV).



Parametry

Parametr	Opis
enable	Sygnal wejściowy. Doprowadzenie tego sygnału powoduje wykonanie bloku funkcyjnego COMMREQ.
IN	Adres pierwszego słowa bloku danych sterujących.
SYSID	Numer kasety (bajt starszy) i numer gniazda (bajt młodszy), w którym jest zainstalowany port przez który ma zostać nawiązana komunikacja.
TASK	Parametr TASK zawiera numer funkcji do wykonania.
FT	Na wejście FT podawany jest sygnał, gdy jednostce centralnej nie udaje się nawiązać komunikacji z określonym modulem.

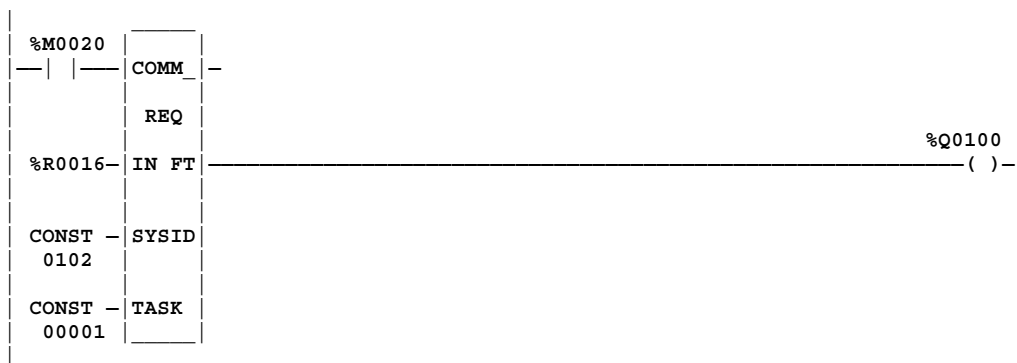
Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN								•	•	•		
SYSID		•	•	•	•		•	•	•	•	•	
TASK								•	•	•	•	
FT	•											•

- Dozwolony typ parametru.

Przykład

W zamieszczonym poniżej przykładzie, po ustawieniu zmiennej %M0020 na 1 następuje wysłanie danych do modułu zainstalowanego w kasecie 1, 2 gnieździe sterownika. Dane te są umieszczone pod adresem %R0016. W przypadku wystąpienia błędu w czasie wykonywania bloku funkcyjnego COMMREQ, wartość zmiennej %Q0100 ustawiana jest na 1.



Uwaga

W przypadku systemów nie mających kaset dodatkowych, dla kasety głównej należy wprowadzić parametr SYSID o wartości zero.

Rozdział 10

Funkcje do operacji tablicowych

Do operacji tablicowych przeznaczone są następujące funkcje:

Oznaczenie skrótowe	Element	Opis	Strona
ARRAY_MOVE	Kopiowanie danych	Kopiowanie określonej liczby danych z tablicy źródłowej do tablicy docelowej	10-1
SRCH_EQ	Szukanie wartości zadanej	Przeszukiwanie tablicy danych w celu znalezienia wartości równej wartości zadanej.	10-6
SRCH_NE	Szukanie wartości różnej	Przeszukiwanie tablicy danych w celu znalezienia wartości różnej od wartości zadanej.	10-6
SRCH_GT	Szukanie wartości większej	Przeszukiwanie tablicy danych w celu znalezienia wartości większej od wartości zadanej.	10-6
SRCH_GE	Szukanie wartości większej lub równej	Przeszukiwanie tablicy danych w celu znalezienia wartości większej lub równej wartości zadanej.	10-6
SRCH_LT	Szukanie wartości mniejszej	Przeszukiwanie tablicy danych w celu znalezienia wartości mniejszej od wartości zadanej.	10-6
SRCH_LE	Szukanie wartości mniejszej lub równej	Przeszukiwanie tablicy danych w celu znalezienia wartości mniejszej lub równej wartości zadanej.	10-6

Maksymalna długość ciągu, będącego parametrem tych funkcji wynosi 32 767 bajtów lub słów, albo 262 136 bitów (operacje na bitach mogą być wykonywane wyłącznie w przypadku funkcji ARRAY_MOVE).

Funkcje te wykonują operacje na następujących typach danych:

Typ	Opis
INT	Liczba całkowita ze znakiem (16 bitowa)
DINT	Liczba całkowita podwójnej precyzji ze znakiem (32 bitowa).
BIT *	Ciąg bitów.
BYTE	Dane typu bajt.
WORD	Dane typu Word (Słowo).

* Dotyczy wyłącznie funkcji ARRAY_MOVE.

Domyślny typ danych to liczba całkowita ze znakiem. Typ danych można zmienić po wprowadzeniu bloku funkcyjnego do szczebla drabiny logicznej programu sterującego. W celu porównania danych innych typów lub danych o różnych typach, należy je uprzednio przekonwertować do jednego z wymienionych typów danych za pomocą odpowiedniej funkcji (proszę porównać z Rozdziałem 11 "Funkcje konwersji").

ARRAY_MOVE (INT, DINT, BIT, BYTE, WORD)

Funkcja ARRAY_MOVE umożliwia skopiowanie określonej liczby elementów z tablicy źródłowej do tablicy docelowej.

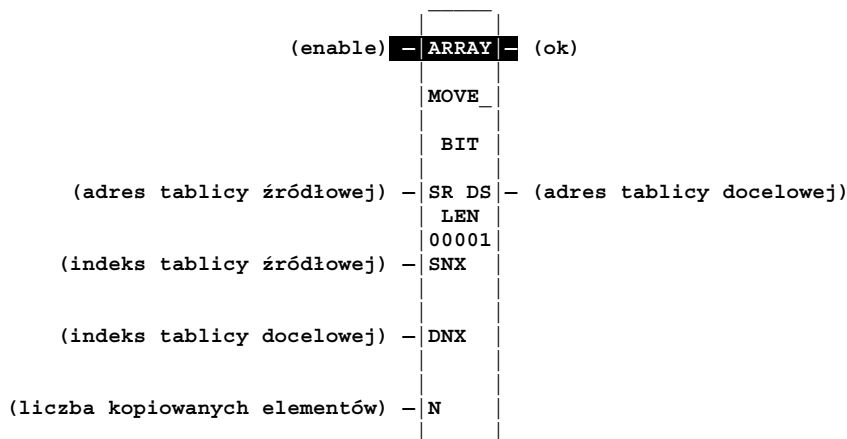
Funkcja ARRAY_MOVE posiada pięć parametrów wejściowych i dwa parametry wyjściowe. Po otrzymaniu sygnału wejściowego, funkcja ta kopiuje N elementów ze źródłowej tablicy danych (rozpoczynającej się od adresu SR), począwszy od "SR + SNX - 1". Elementy te są następnie zapisywane w docelowej tablicy danych, począwszy od "DS + DNX - 1". Parametr LEN określa liczbę elementów w tablicach.

Jeżeli w przypadku funkcji ARRAY_MOVE_BIT (operującej na bitach) podany zostanie adres początkowy tablicy źródłowej i/lub docelowej leżący w obszarze pamięci zorientowanej rejestrowo, pierwszym bitem tablicy jest najmniej znaczący bit wyszczególnionego rejestru. Wyświetlana wartość zawiera 16 bitów, bez względu na długość tablicy.

Wskaźniki tablic funkcji ARRAY_MOVE liczone są od 1. Przy korzystaniu z funkcji ARRAY_MOVE nie można odwoływać się do elementów położonych poza tablicą źródłową lub docelową (określonymi poprzez adres początkowy i długość).

Funkcja przesyła sygnał wyjściowy, chyba że wystąpi jeden z następujących przypadków:

- Do bloku funkcyjnego nie dopływa sygnał wejściowy.
- $(N + SNX - 1)$ jest większe od LEN.
- $(N + DNX - 1)$ jest większe od LEN.



Parametry

Parametr	Opis
enable	Sygnal wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
SR	Adres początkowy tablicy źródłowej. W przypadku funkcji ARRAY_MOVE_BIT można podać dowolną zmienną dyskretną, bez uwzględniania podziału na bajty. Nie mniej jednak, w czasie pracy wyświetlanych jest 16 bitów, począwszy od zadanego adresu.
SNX	Indeks pierwszego z kopiowanych elementów tablicy źródłowej.
DNX	Indeks pierwszego z kopiowanych elementów tablicy docelowej.
N	Liczba elementów do skopiowania.
ok	Sygnal wyjściowy wysyłany po doprowadzeniu sygnału wejściowego.
DS	Adres początkowy tablicy docelowej. W przypadku funkcji ARRAY_MOVE_BIT, można podać dowolną zmienną dyskretną, bez uwzględniania podziału na bajty. Nie mniej jednak, w czasie pracy wyświetlanych jest 16 bitów, począwszy od zadanego adresu.
LEN	Liczba elementów w każdej z tablic.

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
SR		o	o	o	o	Δ†	o	•	•	•		
SNX		•	•	•	•		•	•	•	•	•	
DNX		•	•	•	•		•	•	•	•	•	
N		•	•	•	•		•	•	•	•	•	
Ok	•											•
DS.		o	o	o	o	†	o	•	•	•		

- Dozwolony typ parametru.
W funkcji ARRAY_MOVE_BIT, wartości zmiennych dyskretnych %I, %Q, %M i %T nie muszą być podawane z uwzględnieniem podziału na bajty.
- o Typ może być wykorzystany wyłącznie jako parametr do operacji na danych typu INT, BIT, BYTE lub WORD. Nie można go wykorzystywać w przypadku operacji na danych typu DINT.
- Δ Typ może być wykorzystany wyłącznie jako parametr do operacji na danych typu BIT, BYTE lub WORD. Nie można go wykorzystywać w przypadku operacji na danych typu INT lub DINT.
- † Można stosować wyłącznie zmienne systemowe typu %SA, %SB i %SC. Nie można korzystać ze zmiennych typu %S.

Przykład 1

W zamieszczonym poniżej przykładzie, rejestry %R0003 - %R0007 będące elementami tablicy %R0001 - %R0016 są kopiowane do rejestrów %R0104 - %R0108 tablicy %R0100 - %R0115.

```

%I0001  ARRAY -
-----|----- MOVE_
                WORD
                %R0001- SR DS - %R0100
                LEN
                00016
CONST - SNX
00003
CONST - DNX
00005
CONST - N
00005
    
```

Przykład 2

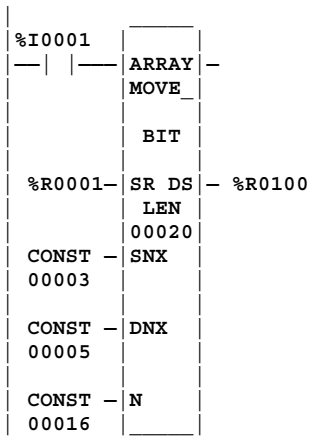
W niniejszym przykładzie, tablice określone adresami SR i DS znajdują się w obszarze pamięci bitowej. Elementy %M0011 - %M0017 tablicy %M009 - %M0024 są zapisywane do elementów %Q0026 - %Q0032 tablicy %Q0022 - %Q0037.

```

%I0001  ARRAY -
-----|----- MOVE_
                BIT
                %M0009- SR DS - %Q0022
                LEN
                00016
CONST - SNX
00003
CONST - DNX
00005
CONST - N
00007
    
```

Przykład 3

Obydwe tablice umieszczone są w obszarze pamięci słów. Odczytywane są bity, począwszy od trzeciego najbardziej znaczącego bitu rejestru %R0001 do drugiego najmniej znaczącego bitu zmiennej %R0002 tablicy zawierającej wszystkie 16 bitów rejestru %R0001 i cztery bity rejestru %R0002. Następnie bity te są zapisywane w miejsce, od piątego najmniej znaczącego bitu rejestru %R0100 do czwartego najmniej znaczącego bitu rejestru %R0101 tablicy zawierającej wszystkie 16 bitów rejestru %R0100 i cztery bity rejestru %R0101.



Funkcje do przeszukiwania tablicy

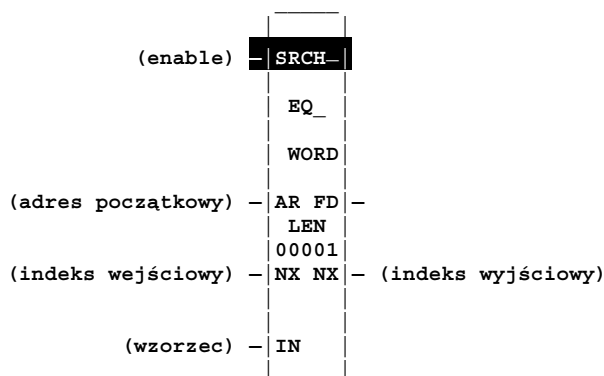
Funkcje te umożliwiają przeszukiwanie tablicy w celu znalezienia wszystkich wartości, spełniających określone kryterium.

Oznaczenie skrótowe	Element	Opis
SRCH_EQ	Szukanie wartości zadanej	Przeszukiwanie tablicy danych w celu znalezienia wartości równej wartości zadanej.
SRCH_NE	Szukanie wartości różnej	Przeszukiwanie tablicy danych w celu znalezienia wartości różnej od wartości zadanej.
SRCH_GT	Szukanie wartości większej	Przeszukiwanie tablicy danych w celu znalezienia wartości większej od wartości zadanej.
SRCH_GE	Szukanie wartości większej lub równej	Przeszukiwanie tablicy danych w celu znalezienia wartości większej lub równej wartości zadanej.
SRCH_LT	Szukanie wartości mniejszej	Przeszukiwanie tablicy danych w celu znalezienia wartości mniejszej od wartości zadanej.
SRCH_LE	Szukanie wartości mniejszej lub równej	Przeszukiwanie tablicy danych w celu znalezienia wartości mniejszej lub równej wartości zadanej.

Każda z tych funkcji posiada cztery parametry wejściowe i dwa parametry wyjściowe. Po doprowadzeniu do funkcji sygnału, następuje rozpoczęcie przeszukiwania tablicy, począwszy od adresu "AR + NX" (początek tablicy plus indeks pierwszego elementu danych podlegających porównaniu, czyli parametr wejściowy NX).

Poszukiwanie jest kontynuowane do momentu znalezienia pierwszego elementu spełniającego zadany warunek lub do momentu dojścia do końca tablicy. Jeśli poszukiwany element zostanie znaleziony, wartość parametru wyjściowego FD zostanie ustawiona na 1, a na wyjście oznaczone NX skopiowany zostanie indeks elementu, spełniającego warunek poszukiwań. Jeśli w przeszukiwanym obszarze nie zostanie znaleziony element spełniający warunek poszukiwań, wartości obydwu parametrów wyjściowych FD i NX są ustawiane na 0.

Parametr wejściowy NX może przyjmować wartości z zakresu od 0 do "LEN - 1". Aby rozpocząć przeszukiwanie tablicy od pierwszego elementu wartość tego parametru powinna wynosić 0. Podczas wykonywania operacji przeszukiwania tablicy wartość ta jest zwiększana o 1. Tak więc wartości parametru wyjściowego NX mogą zawierać się w przedziale od 1 do LEN. Jeśli wartość parametru wejściowego NX wychodzi poza zakres (jest mniejsza od 0 lub większa lub równa od LEN), jest ona ustawiana samoczynnie na 0.



Parametry

Parametr	Opis
enable	Sygnal wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
AR	Adres początkowy tablicy danych, która ma zostać przeszukana.
Indeks wejściowy NX	Indeks pierwszego z porównywanych elementów.
IN	Kryterium do przeszukiwania (wartość, z którą poszukiwany element ma pozostawać w określonej relacji).
Indeks wyjściowy NX	Indeks (pozycja w tablicy) znajdującego elementu pozostającego w określonej relacji z elementami porównawczym.
FD	Wskaźnik informujący czy poszukiwany element został znaleziony.
LEN	Liczba elementów tworzących tablicę podlegającą przeszukiwaniu. Wartość ta musi mieścić się w zakresie od 1 do 32767 bitów lub słów.

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
AR		o	o	o	o	Δ	o	•	•	•		
NX wejściowe		•	•	•	•		•	•	•	•	•	
IN		o	o	o	o	Δ	o	•	•	•	•	
NX wyjściowe		•	•	•	•		•	•	•	•		
FD	•											•

- Typ może być wykorzystany jako dany parametr bloku funkcyjnego.
- o Typ może być wykorzystany wyłącznie jako parametr do operacji na danych typu INT, BYTE lub WORD. Nie można go wykorzystywać w przypadku operacji na danych typu DINT.
- Δ Typ może być wykorzystany wyłącznie jako parametr do operacji na danych typu BYTE lub WORD. Nie można go wykorzystywać w przypadku operacji na danych typu INT lub DINT.

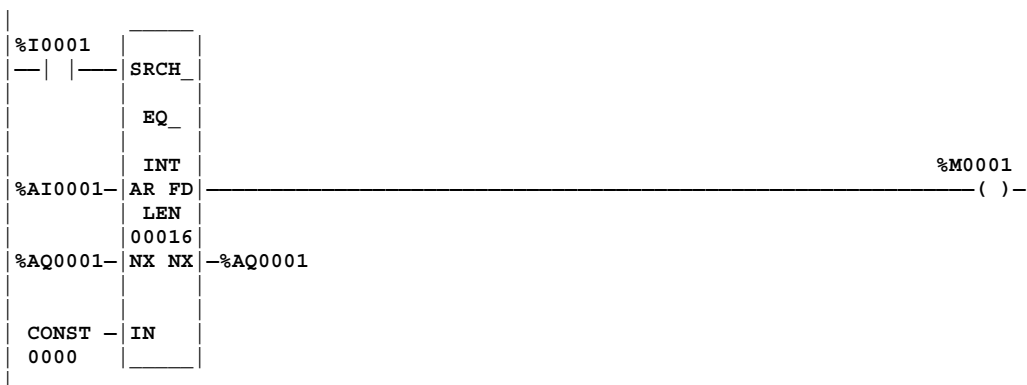
Przykład 1

Tablica AR przechowywana jest w pamięci określonej adresami %R0001 - %R0005. Po ustawieniu wartości parametru EN na 1, część tablicy, określona adresami %R0004 do %R0005 jest przeszukiwana w celu znalezienia elementu którego wartość jest równa parametrowi IN. Jeżeli %R0001 = 7, %R0002 = 9, %R0003 = 6, %R0004 = 7, %R0005 = 7 i %R0100 = 7, poszukiwanie zostanie rozpoczęte od %R0004 i zakończone w momencie gdy wartość parametr FD zostanie ustawiony na 1, w %R0004, a do %R0101 zostanie zapisana wartość 4.



Przykład 2

Tablica AR przechowywana jest w pamięci określonej adresami %AI0001 - %AI0016. Elementy tej tablicy są równe odpowiednio 100, 20, 0, 5, 90, 200, 0, 79, 102, 80, 24, 34, 987, 8, 0, 500. Początkowo indeks %AQ0001 jest równy 5. Tablica jest przeszukiwana w każdym cyklu w celu znalezienia elementu o wartości równej wartości parametru IN, wynoszącej 0. W pierwszym cyklu, poszukiwanie rozpoczęte zostanie od adresu %AI0006, a zakończone w %AI0007, taki więc wartość FD jest wtedy równa 1, a wartość %AQ0001 jest równa 7. W drugim cyklu, poszukiwanie zostanie rozpoczęte od adresu %AI0008, a zakończone w %AI0015, tak więc wartość FD będzie równa 1, a wartość %AQ0001 będzie równa 15. Poszukiwanie w następnym cyklu zostanie rozpoczęte od adresu %AI0016. Ponieważ nie zostanie znaleziony szukany element, parametr FD zostanie ustawiony na 0, a wartość %AQ0001 również będzie ustawiona na 0. W kolejnym cyklu, poszukiwanie zostanie rozpoczęte od początku tablicy.



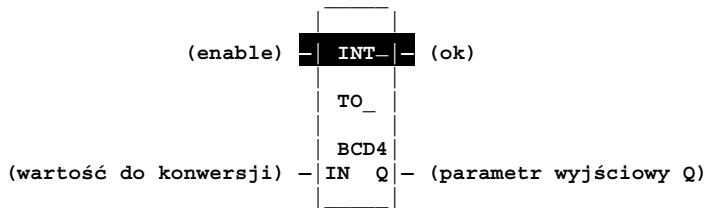
Funkcje te umożliwiają konwersję danych. Konieczność taka wynika z faktu, że wiele bloków funkcyjnych musi operować na danych tego samego typu (jak np. funkcje matematyczne). W rozdziale tym opisano następujące funkcje do konwersji:

Oznaczenie skrótowe	Element	Opis	Strona
BCD-4	Konwersja na kod BCD	Konwersja danych typu INT na kod BCD.	11-2
INT	Konwersja na dane typu INT	Konwersja danych typu BCD-4 na dane typu INT.	11-3
DINT	Konwersja na dane typu DINT	Konwersja danych typu REAL na dane typu DINT.	11-5
REAL	Konwersja na dane typu REAL	Konwersja danych typu INT, DINT, BCD-4 lub WORD na dane typu REAL.	11-7
WORD	Konwersja na dane typu WORD	Konwersja danych typu REAL na dane typu WORD	11-9
TRUN	Przybliżanie z odrzucaniem części dziesiętnej	Przybliżanie liczby rzeczywistej poprzez odrzucenie części dziesiętnej.	11-11

—>BCD-4 (INT)

Blok funkcyjny przeznaczony jest do konwersji danych zapisanych jako liczba całkowita ze znakiem (typ INT) na równoważną liczbę zapisaną w kodzie BCD-4 (liczba dziesiętna zakodowana w układzie dwójkowym). Dane wejściowe nie ulegają zmianie. Konwersja taka może być wymagana np. w celu podłączenia wyświetlacza sterowanego w kodzie BCD lub w celu przesłania kodów sterujących do niektórych urządzeń zewnętrznych, jak np. do licznika impulsów o wysokiej częstotliwości.

Po doprowadzeniu sygnału wejściowego przeprowadzona zostaje konwersja wartości zadanej parametrem IN, a jej wynik jest zapisywany do parametru wyjściowego Q. Sygnał wyjściowy jest zawsze przesyłany po doprowadzeniu sygnału wejściowego, chyba że po dokonaniu konwersji wartość parametru Q przekracza zakres od 0 do 9999.



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
IN	Wartość stała lub adres zmiennej, której wartość podlega konwersji na system zapisu BCD-4.
ok	Sygnał wyjściowy, wysyłany po poprawnym wykonaniu funkcji.
Q	Wartość parametru wejściowego IN po dokonaniu konwersji.

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN		•	•	•	•		•	•	•	•	•	
ok	•											•
Q		•	•	•	•		•	•	•	•		

- Dozwolony typ parametru.

Przykład

W zamieszczonym poniżej przykładzie, po każdej zmianie wartości zmiennej %I0002 na 1, jeżeli nie wystąpiły żadne błędy, parametry wejściowe z obszaru %I0017 do %I0032 są konwertowane na liczby w kodzie BCD, a wynik zapisywany jest w pamięci %Q0033 do %Q0048. Poprawność wykonania konwersji jest sprawdzana za pomocą przekaźnika %Q1432.



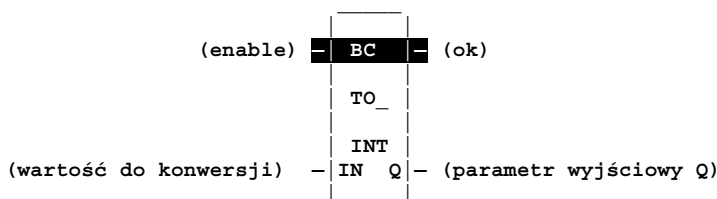
—>INT (BCD-4, REAL)

Blok funkcyjny stosowany jest do konwersji danych typu BCD-4 lub REAL na równoważną liczbę całkowitą ze znakiem (INT). Dane wejściowe nie ulegają zmianie.

Uwaga

Operacje na liczbach rzeczywistych można realizować wyłącznie w jednostkach centralnych 35x i 36x, oprogramowanie systemowe ver. 9 lub nowsza oraz we wszystkich wersjach jednostki centralnej CPU352.

Po doprowadzeniu sygnału wejściowego przeprowadzana jest konwersja wartości zadanej parametrem IN, a wynik zapisywany jest do parametru wyjściowego Q. Sygnał wyjściowy jest zawsze przesyłany po doprowadzeniu sygnału wejściowego, chyba że dane przekraczają zakres wartości dopuszczalnych.



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
IN	Wartość stała albo adres zmiennej typu BCD lub REAL, której wartość podlega konwersji na typ INT.
ok	Sygnał wyjściowy wysyłany po doprowadzeniu sygnału wejściowego i dokonania konwersji bez wystąpienia przekroczenia dozwolonego zakresu wartości.
Q	Wartość parametru wejściowego IN po dokonaniu konwersji.

Dopuszczalne typy parametrów

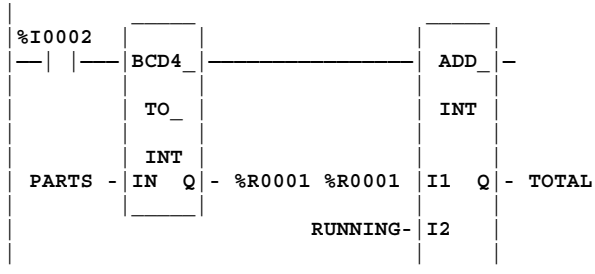
Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN		•	•	•	•		•	•	•	•	•	
ok	•											•
Q		•	•	•	•		•	•	•	•		

Uwaga: W przypadku danych REAL, dopuszczalne jest stosowanie wyłącznie danych typu %R, %AI i %AQ.

- Dozwolony typ parametru.

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości parametru wejściowego %I0002 na 1, powoduje konwersję parametru wejściowego PARTS typu BCD-4 na liczbę całkowitą ze znakiem, która następnie jest przesyłana do funkcji ADD i dodawana do parametru RUNNING typu liczba całkowita ze znakiem. Suma tych dwóch liczb jest zwracana przez funkcję ADD jako parametr TOTAL.



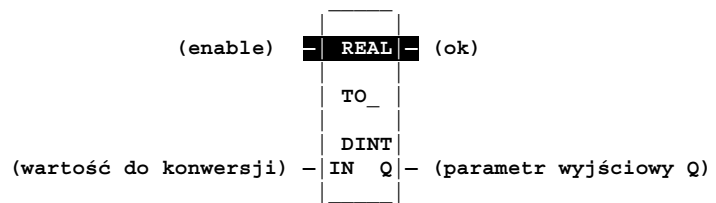
—>DINT (REAL)

Funkcja ta umożliwia konwersję danych typu REAL na dane typu DINT. Dane wejściowe nie ulegają zmianie.

Uwaga

Operacje na liczbach rzeczywistych można realizować wyłącznie w jednostkach centralnych 35x i 36x, oprogramowanie systemowe wer. 9 lub nowsza oraz we wszystkich wersjach jednostki centralnej CPU352.

Po doprowadzeniu sygnału wejściowego przeprowadzana jest konwersja wartości zadanej parametrem IN, a wynik zapisywany jest do parametru wyjściowego Q. Sygnał wyjściowy jest zawsze przesyłany po doprowadzeniu sygnału wejściowego, chyba że wartość liczby rzeczywistej przekracza zakres dopuszczalnych wartości.



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
IN	Wartość stała lub adres zmiennej, której wartość podlega konwersji typ DINT.
ok	Sygnał wyjściowy wysyłany po doprowadzeniu sygnału wejściowego i dokonaniu konwersji, o ile liczba rzeczywista mieści się w dozwolonym zakresie wartości.
Q	Wartość parametru wejściowego IN po dokonaniu konwersji.

Uwaga

W przypadku konwersji danych typu REAL na dane typu DINT w wyniku może zostać otrzymana liczba o mniejszej dokładności, ponieważ dane typu REAL posiadają 24 bity znaczące.

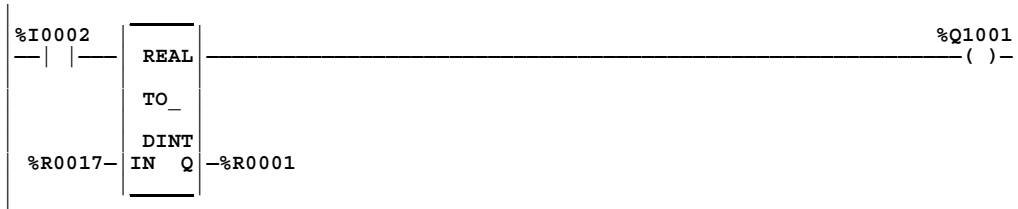
Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN		o	o	o	o		o	•	•	•	•	
ok	•											•
Q								•	•	•		

- Dozwolony typ parametru.

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0002 na 1 powoduje konwersję parametru wejściowego %R0017 na liczbę całkowitą podwójnej precyzji ze znakiem i zapisanie wyniku w parametrze wyjściowym %R0001. Wartość parametru wyjściowego %Q1001 jest zawsze ustawiana na 1, pod warunkiem pomyślnego wykonania funkcji.



—>REAL (INT, DINT, BCD-4, WORD)

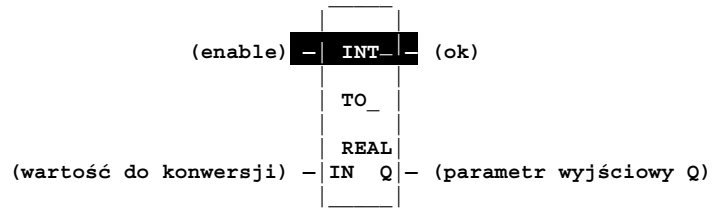
Blok funkcyjny stosowany jest do konwersji danych typu INT, DINT, BCD-4 lub WORD na równoważną liczbę rzeczywistą (REAL). Dane wejściowe nie ulegają zmianie.

Po doprowadzeniu sygnału wejściowego przeprowadzana jest konwersja wartości zadanej parametrem IN, a wynik zapisywany jest do parametru wyjściowego Q. Sygnał wyjściowy jest zawsze przesyłany po doprowadzeniu sygnału wejściowego, chyba że po dokonaniu konwersji wynik nie mieści się w dopuszczalnym zakresie wartości.

W przypadku konwersji danych typu DINT na dane typu REAL w wyniku może zostać otrzymana liczba o mniejszej dokładności, ponieważ liczba bitów znaczących zostaje zredukowana do 24.

Uwaga

Funkcja ta jest dostępna wyłącznie w jednostkach centralnych 35x i 36x, oprogramowanie systemowe wer. 9 lub nowsza oraz we wszystkich wersjach jednostek centralnych CPU352.



Parametry

Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
IN	Wartość stała lub adres zmiennej, której wartość podlega konwersji na typ REAL.
ok	Sygnał wyjściowy, wysyłany po poprawnym wykonaniu funkcji.
Q	Wartość parametru wejściowego IN po dokonaniu konwersji.

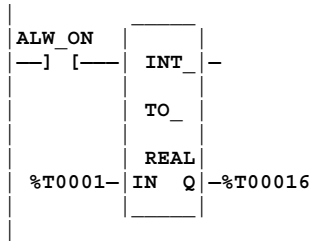
Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN		o	o	o	o		o	•	•	•	•	
ok	•											•
Q								•	•	•		

- Typ może być wykorzystany jako dany parametr bloku funkcyjnego.
- o Zmienna danego typu nie może być zastosowana w przypadku konwersji typu DINT na REAL.

Przykład

W zamieszczonym poniżej przykładzie, wartość parametru wejściowego IN wynosi 678. Zmienna %R0016, w której przechowywany jest wynik, ma wartość 678.000.



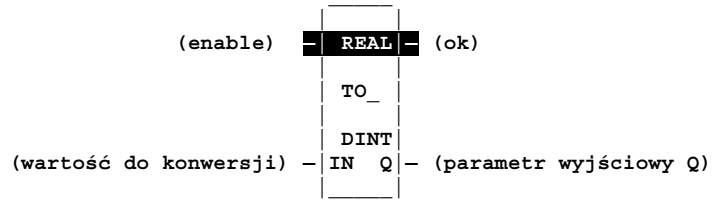
—>WORD (REAL)

Blok ten stosowany jest do konwersji danych typu REAL na równoważną liczbę typu WORD. Dane wejściowe nie ulegają zmianie.

Uwaga

Funkcja ta dostępna jest wyłącznie w jednostkach centralnych serii 35x i 36x.

Po doprowadzeniu sygnału wejściowego przeprowadzana jest konwersja wartości zadanej parametrem IN, a wynik zapisywany jest do parametru wyjściowego Q. Sygnał wyjściowy jest zawsze przesyłany po doprowadzeniu sygnału wejściowego, chyba że po dokonaniu konwersji wartość parametru Q przekracza zakres od 0 do FFFFh.



Parametry

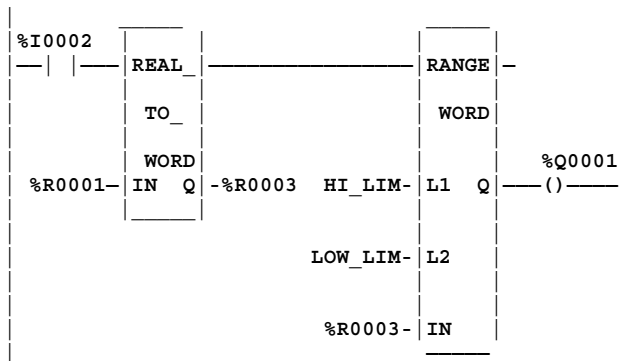
Parametr	Opis
enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
IN	Wartość stała lub adres zmiennej, której wartość podlega konwersji na typ WORD.
ok	Sygnał wyjściowy, wysyłany po poprawnym wykonaniu funkcji.
Q	Wartość parametru wejściowego IN po dokonaniu konwersji.

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN								•	•	•	•	
ok	•											•
Q		•	•	•	•		•	•	•	•		

- Dozwolony typ parametru.

Przykład



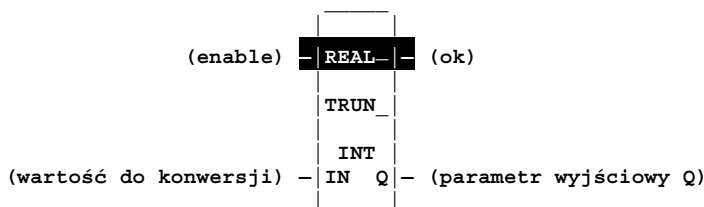
TRUN (INT, DINT)

Funkcja ta umożliwia zaokrąglenie liczby rzeczywistej poprzez odrzucenie części dziesiętnej. Dane wejściowe nie ulegają zmianie.

Uwaga

Operacje na liczbach rzeczywistych mogą realizować wyłącznie jednostki centralne 35x i 36x (oprogramowanie systemowe ver. 9 i nowsze oraz wszystkie wersje jednostek centralnych CPU352), a więc funkcja TRUN nie jest obsługiwane przez pozostałe jednostki centralne serii 90-30.

Po doprowadzeniu sygnału wejściowego przeprowadzana jest konwersja wartości zadanej parametrem IN, a wynik zapisywany jest do parametru wyjściowego Q. W przypadku jednostki centralnej CPU352, funkcja zawsze przesyła sygnał wyjściowy po doprowadzeniu sygnału wejściowego, chyba że po dokonaniu konwersji wynik nie mieści się w dopuszczalnym zakresie wartości lub parametr IN ma wartość NaN (Not a Number). W przypadku innych jednostek centralnych serii 35x i 36x, funkcja *nie przesyła* sygnału wyjściowego.



Parametry

Parametr	Opis
Enable	Sygnał wejściowy. Doprowadzenie tego sygnału powoduje wykonanie żądanej operacji.
IN	Stała lub zmienna, której wartość ma zostać zaokrąglona.
Ok	Sygnał wyjściowy wysyłany, jeżeli działanie zostało poprawnie wykonane, jego wynik mieści się w dopuszczalnym przedziale wartości a parametr wejściowy IN jest liczbą.
Q	Wartość parametru wejściowego IN po dokonaniu konwersji.

Uwaga

W przypadku konwersji danych typu REAL na dane typu DINT w wyniku może zostać otrzymana liczba o mniejszej dokładności, ponieważ dane typu REAL posiadają 24 bity znaczące.

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
IN								•	•	•	•	
ok	•											•
Q		o	o	o	o		o	•	•	•		

- Typ może być wykorzystany jako dany parametr bloku funkcyjnego.
- o Wyłącznie w przypadku zaokrąglania danych typu REAL do danych typu INT.

Przykład

W zamieszczonym poniżej przykładzie następuje zaokrąglenie zadanej stałej, a wynik równy 562 jest zwracany poprzez zmienną %T0001.

```

ALW_ON
---] [----- REAL ---
                TRUN
                INT
CONST - IN Q  -%T0001
5.62987E+02

```

Rozdział 12

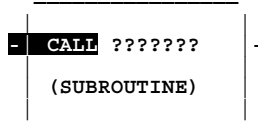
Funkcje sterujące

W niniejszym rozdziale opisano funkcje sterujące wykonywaniem programu, umożliwiające zmianę sposobu jego działania lub wprowadzenie pewnych ograniczeń. (Cykle pracy sterownika opisano w Rozdziale 2 "Cykl pracy sterownika").

Funkcja	Opis	Strona
CALL	Wywołanie podprogramu.	12-2
DOIO	Natychmiastowe uaktualnienie stanu wybranych wejść lub wyjść na okres jednego cyklu pracy sterownika. (Funkcja ta obsługuje wszystkie wejścia i wyjścia modułu, wyszczególnione jako parametry jej wywołania). Nie jest możliwe uaktualnienie wybranych (częściowe) wejść i wyjść modułu. Opcjonalnie można umieścić kopię obsługiwanych wejść i wyjść w pamięci wewnętrznej, a nie w standardowej pamięci wejść dyskretnych	12-3
SER	Blok funkcyjny SER (Sequential Event Recorder) przeznaczony jest do rejestrowania serii próbek. Sposób wykonywania tego bloku funkcyjnego uzależniony jest od parametrów, definiowanych przez użytkownika w bloku danych sterujących.	12-8
END	koniec programu. Program wykonywany jest od pierwszego szczebla drabiny logicznej aż do ostatniego szczebla, jeżeli jednak napotkana zostanie instrukcja END, wykonanie programu zostaje bezwarunkowo przerwane. Funkcja END jest użyteczna podczas uruchamiania programu, nie jest jednak dozwolone korzystanie z niej przy programowaniu w języku SFC (proszę porównać z uwagą na stronie 12-8).	12-21
MCR i MCRN	Przełącznik sterujący wykonywaniem fragmentu programu. Blok funkcyjny MCR wyznacza początek fragmentu programu sterującego, w którym wszystkie szczeble zostaną wykonane bez dopływu sygnału sterującego. Koniec takiego fragmentu wyznaczony jest przez odpowiedni blok funkcyjny ENDMCR. Oprogramowanie Logicmaster 90-30/20/Micro udostępnia dwa rodzaje funkcji MCR: z możliwością zagnieżdżania (MCRN) i bez możliwości zagnieżdżania (MCR).	12-22
ENDMCR i ENDMCRN	Instrukcja ta odwołuje wcześniejszą instrukcję MCR (MCRN). Oprogramowanie Logicmaster 90-30/20/Micro udostępnia dwa rodzaje funkcji ENDMCR: z możliwością zagnieżdżania (ENDMCRN) i bez możliwości zagnieżdżania (ENDMCR).	12-25
JUMP i JUMPN	Przejdzie do innego miejsca w programie sterującym (oznaczonego instrukcją LABEL). Oprogramowanie Logicmaster 90-30/20/Micro udostępnia dwa rodzaje funkcji JUMP: bez możliwości pokrywania się zakresów działania (JUMP) i z możliwością pokrywania się zakresów działania (JUMPN).	12-26
LABEL i LABELN	Miejsce docelowe dla instrukcji skoku (JUMP). Oprogramowanie Logicmaster 90-30/20/Micro udostępnia dwa rodzaje funkcji LABEL: bez zagnieżdżania (LABEL) i z możliwością zagnieżdżania (LABELN).	12-28
COMMENT	Wstawienie komentarza (objaśnienia danego szczebla programu sterującego). W celu wprowadzenia zmian w tekście komentarza należy ustawić na nim kursor, a następnie wcisnąć F10 (Zoom).	12-30
SVCREQ	Wywołanie funkcji specjalnej sterownika. (Proszę porównać z listą bloków funkcyjnych SVCREQ, zamieszczono na stronie 12-30.)	12-29
PID	Regulatory proporcjonalno- całkowo- różniczkowe: <ul style="list-style-type: none"> • Standardowy regulator PID ISA (PIDISA). • Regulator PID o niezależnych wyrazach (PIDIND). 	12-64

CALL

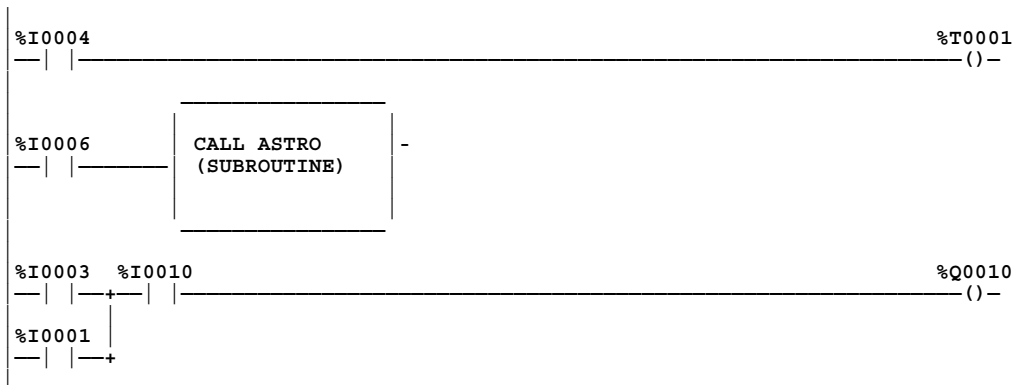
Blok funkcyjny CALL umożliwia wywołanie podprogramu w danym miejscu programu sterującego.



Po doprowadzeniu do bloku funkcyjnego CALL sygnału wejściowego następuje natychmiastowe wywołanie i wykonanie żądanego podprogramu. Po zakończeniu wykonywania podprogramu, sygnał powraca do szczebla umieszczonego bezpośrednio po bloku CALL.

Przykład

W zamieszczonym poniżej przykładzie następuje wywołanie podprogramu. Ustawienie kursora wewnątrz instrukcji CALL i naciśnięcie klawisza **F10** powoduje przejście do podprogramu.



Uwaga

Sterowniki Micro nie udostępniają możliwości korzystania z podprogramów. Nie można więc, w przypadku tych sterowników, korzystać z instrukcji CALL.

DOIO

Funkcja DOIO stosowana jest w celu natychmiastowego uaktualnienia stanu wybranych wejść lub wyjść podczas wykonywania programu sterującego, w czasie trwania cyklu pracy sterownika. Funkcja ta umożliwia dodatkowe uaktualnienie wybranych wejść/wyjść, poza normalną obsługą wejść i wyjść, wykonywaną odpowiednio na początku i przy końcu każdego z cykli pracy sterownika.

W przypadku wyszczególnienia jako parametrów zmiennych przypisanych wejściom sterownika, funkcja DOIO umożliwia odczytanie z poziomu programu sterującego najbardziej aktualnych wartości fizycznych wejść sterownika. Jeśli jako parametry podane są zmienne przypisane wyjściom sterownika, nastąpi uaktualnienie stanu wyjść. Wejścia i wyjścia są obsługiwane przyrostowo dla całych modułów wejść/wyjść, jeżeli to jest konieczne, sterownik uaktualnia zmienne w trakcie wykonywania funkcji.

Funkcja DOIO posiada cztery parametry wejściowe i jeden parametr wyjściowy. Po doprowadzeniu sygnału wejściowego, jeżeli w bloku parametrów wyszczególniono wejścia, których stan ma zostać uaktualniony, odczytany zostaje zbiór wejść o adresie początkowym ST i adresie końcowym END. Dalsze działanie funkcji uzależnione jest od tego, czy wykorzystywany jest parametr ALT (adres). Jeśli podano parametr ALT, uaktualnione zostają wartości zmiennych, rozpoczynając od wyspecyfikowanego za pomocą parametru ALT adresu (może to być na przykład obszar pamięci przypisany zmiennym typu %M lub %R), a standardowa pamięć sygnałów wejść dyskretnych nie jest uaktualniana. Typ parametru ALT musi być taki sam jak typ aktualizowanej zmiennej. Jeśli jako parametry ST i END podano zmienne dyskretne, parametr ALT musi być również zmienną dyskretną. Jeśli nie podano parametru ALT, uaktualnione zostają rzeczywiste wartości zmiennych przypisanych skanowanym wejściom (tzn. %I lub %AI).

Po doprowadzeniu sygnału wejściowego, jeżeli w bloku parametrów wyszczególniono wyjścia, do modułów wyjść przesłane zostają wartości zmiennych przypisanych blokowi wyjść o adresie początkowym ST i adresie końcowym END. Jeśli na wyjścia sterownika mają zostać przesłane wartości z innego obszaru pamięci niż %Q czy %AQ, początkowy adres tego obszaru musi zostać podany za pomocą parametru ALT. Wartości z tego obszaru są następnie przesyłane do zbioru wyjść adresowanego przez parametry ST i END.

Funkcja DOIO jest wykonywana do momentu, aż wszystkie wyszczególnione wejścia lub wyjścia zostaną obsłużone. Po wykonaniu tej funkcji program sterujący przechodzi do kolejnej instrukcji.

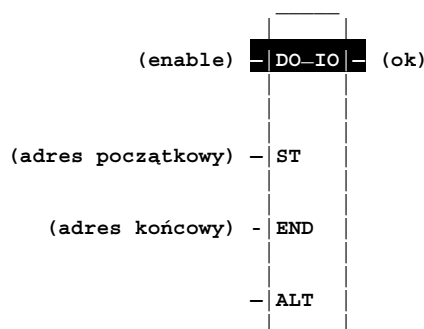
Jeżeli wyszczególnione zmienne obejmują moduły opcjonalne (HSC, APM, itp.), obsłużone zostaną wszystkie dane wejściowe (%I i %AI) i wszystkie dane wyjściowe (%Q i %AQ) dla tego modułu. Parametr ALT jest ignorowany przy obsłudze modułów wyspecjalizowanych. Wyszczególniony zakres zmiennych nie może obejmować modułu Enhanced GCM (proszę porównać z Uwagą poniżej).

Uwaga

W przypadku jednostek centralnych z oprogramowaniem systemowym wer. 9.0 i nowszym, funkcja DOIO może współpracować z modułem komunikacyjnym Enhanced GCM.

Sygnal wyjściowy jest wysyłany zawsze, po doprowadzeniu sygnału wejściowego, o ile nie wystąpi jeden z podanych poniżej przypadków:

- Nie wszystkie zmienne wyszczególnionego typu, z zadanego obszaru pamięci, są fizycznie zrealizowane (np. brakuje modułu wejść lub wyjść, którym przypisane są zmienne wyszczególnione jako parametry funkcji).
- Jednostka centralna sterownika nie jest w stanie zrealizować instrukcji.
- Obszar pamięci wyszczególniony za pomocą parametrów bloku zawiera moduły, których brak został zgłoszony w postaci komunikatu alarmowego "Brak modułu wejść/wyjść".



Parametry

Parametr	Opis
enable	Sygnal wejściowy. Doprowadzenie tego sygnału powoduje uaktualnienie stanu wejść/wyjść sterownika.
ST	Adres początkowy zbioru wejść/wyjść, które mają być uaktualnione.
END	Adres końcowy zbioru wejść/wyjść, które mają być uaktualnione.
ALT	W przypadku odczytu wejść, parametr ten zawiera adres początkowy pamięci sterownika, gdzie zapisany ma być stan zbioru wejść. W przypadku ustawiania wyjść, parametr ALT określa adres obszaru w pamięci sterownika, z którego poszczególne wartości mają zostać skopiowane na odpowiadające im fizyczne wyjścia sterownika. W jednostkach centralnych model 331 i nowszych, parametr ALT może wpływać na prędkość wykonywania funkcji DOIO (porównać z zamieszczoną poniżej Uwagą oraz punktem poświęconym rozszerzonej funkcji DO/IO, dostępnej w jednostkach centralnych model 331 i późniejszych, strona 12-7).
ok	Sygnal wyjściowy wysyłany po pomyślnym wykonaniu funkcji.

Uwaga

W przypadku jednostek centralnych model 331 i późniejsze, parametr ALT funkcji DOIO może być wykorzystany do adresowania gniazda pojedynczego modułu w kasecie głównej. Jeżeli wykorzystana zostanie taka możliwość, wykonanie funkcji DOIO będzie trwać 80 mikrosekund, a nie 236 mikrosekund, jak to ma miejsce w przypadku nie wykorzystywania parametru ALT. Nie ma żadnego mechanizmu do sprawdzania pokrywania się adresów zmiennych lub do badania niezgodności typów modułów.

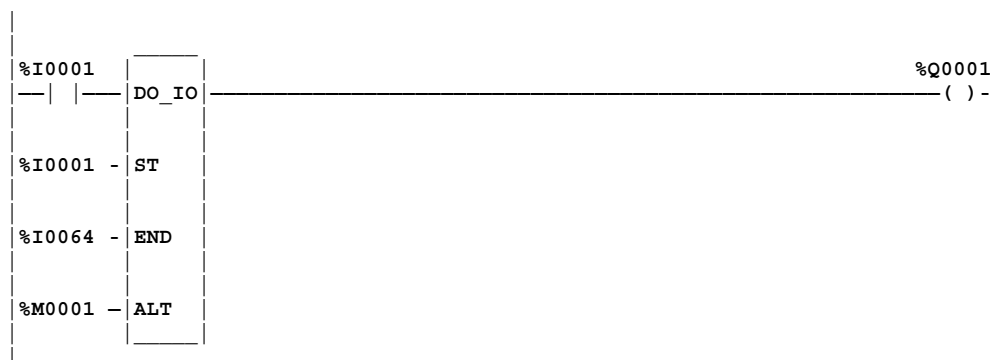
Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
ST		•	•						•	•		
END		•	•						•	•		
ALT		•	•	•	•		•	•	•	•		•
ok	•											•

- Dozwolony typ parametru.

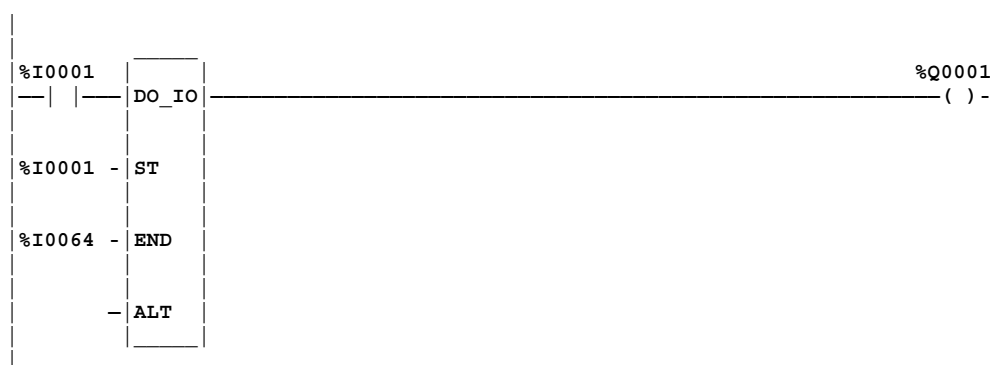
Uaktualnianie stanu wejść, przykład 1

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0001 na 1, powoduje odczytanie zmiennych od %I0001 do %I0064 oraz ustawienie wartości zmiennej %Q0001 na 1. Kopia odczytanych wejść umieszczona zostanie w pamięci wewnętrznej, w obszarze o adresach od %M0001 do %M0064. Standardowa pamięć wejść nie jest aktualizowana. Takie zastosowanie tej funkcji pozwala na porównanie bieżących wartości zbioru wejść z wartościami wejść odczytanymi na początku cyklu.



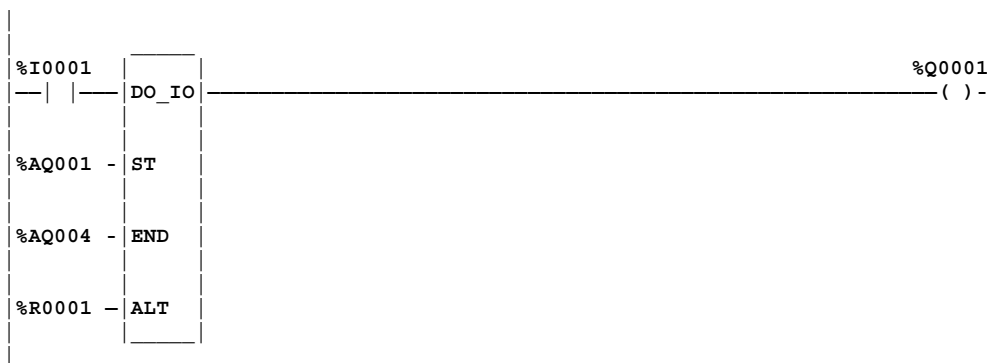
Uaktualnianie stanu wejść, przykład 2

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0001 na 1, powoduje odczytanie zmiennych od %I0001 do %I0064 oraz ustawienie wartości zmiennej %Q0001 na 1. Odczytany stan wejść umieszczany jest w pamięci wejść, w obszarze od %I0001 do %I0064. Takie zastosowanie funkcji pozwala na wielokrotne odczytywanie stanu wejść w części cyklu jednostki centralnej, przeznaczonym na wykonywanie programu sterującego.



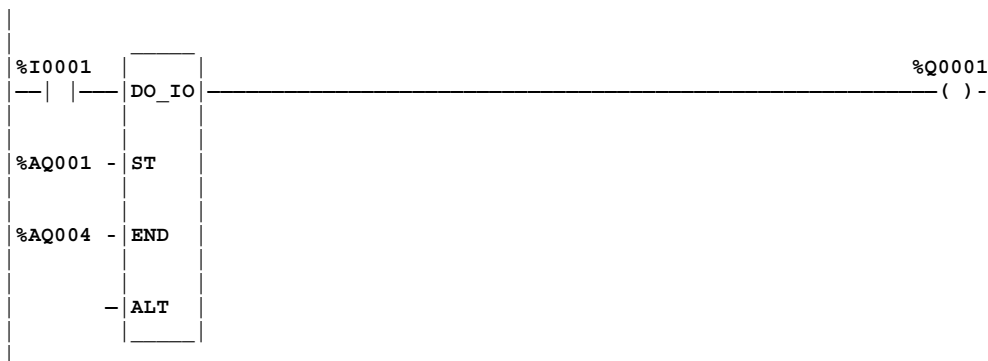
Uaktualnianie stanu wyjść, przykład 1

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0001 na 1 powoduje zapisanie wartości zmiennych %R0001 do %R0004 do wyjść analogowych %AQ001 do %AQ004 oraz ustawienie wartości zmiennej %Q0001 na 1. Wartości zawarte w %AQ001 do %AQ004 nie są zapisywane do modułów wyjść analogowych.



Uaktualnianie stanu wyjść, przykład 2

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0001 na 1 powoduje zapisanie wartości zmiennych %AQ0001 do %AQ0004 do kanałów wyjść analogowych %AQ001 do %AQ004 oraz ustawienie wartości zmiennej %Q0001 na 1.



Rozszerzona funkcja DO I/O dla jednostek centralnych 331 i nowszych

Ostrzeżenie

Jeżeli w programie sterującym wykorzystywana jest rozszerzona funkcja DO I/O, program nie powinien być wczytywany przez oprogramowanie Logimaster 90-30, wersje wcześniejsze od 4.01.

Rozszerzona funkcja DO I/O (DOIO) dostępna jest w jednostkach centralnych z oprogramowaniem systemowym ver. 4.30 lub nowsza, modele 331 i późniejsze. Ta rozszerzona funkcja DOIO może być wykorzystywana wyłącznie dla modułów z pojedynczym modułem wejść/wyjść, 8 punktowym, 16 punktowym lub 32 punktowym.

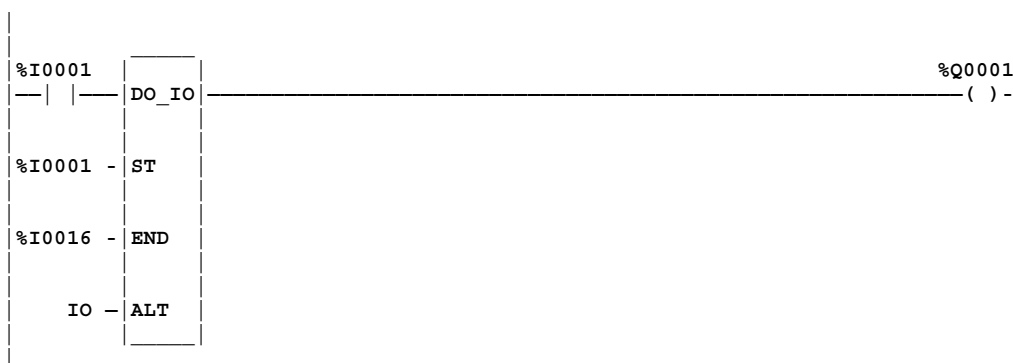
Przy pomocy parametru ALT identyfikowane jest gniazdo w kasecie głównej, w której zainstalowany jest moduł. Przykładowo, jeżeli parametr ten jest stałą o wartości 2, oznacza to, że jednostka centralna wykona rozszerzoną funkcję DOIO dla modułu zainstalowanego w gnieździe 2.

Uwaga

Rozszerzona funkcja DOIO sprawdza jedynie pod kątem błędów stan modułu w określonym gnieździe.

Rozszerzona funkcja DOIO może być wykorzystana wyłącznie do obsługi modułów zainstalowanych w kasecie głównej. Z tego powodu, parametr ALT musi mieć wartość od 2 do 5 dla kaset z 5 gniazdami lub 2 do 10 dla kaset z 10 gniazdami.

Zmienne zawierające adresy początkowy i końcowy muszą być typu %I lub %Q. Zmienne te określają początek i koniec zbioru wejść/wyjść, dla których moduł jest skonfigurowany. Przykładowo, jeżeli moduł z 16 wejściami przypisanymi zmiennym %I0001 do %I0016, zainstalowany jest w 10 gnieździe kasy głównej z 10 gniazdami, parametr ST musi być równy %I0001, parametr END %I0016, a parametr ALT musi mieć wartość równą 10 (tzn. 0AH heksadecymalnie), zgodnie z pokazanym poniżej przykładem.



W zamieszczonej poniżej tabeli porównano czasy wykonania normalnej funkcji DOIO dla modułu z 8, 16 lub 32 wejściami/ wyjściami, z czasami wykonania dla rozszerzonej funkcji DOIO.

Moduł	Normalna funkcja DOIO Czas wykonywania	Rozszerzona funkcja DOIO Czas wykonywania
Moduł wejść dyskretnych, 8 punktowy	224 mikrosekund	67 mikrosekund
Moduł wyjść dyskretnych, 16 punktowy	47 mikrosekund	48 mikrosekund
Moduł wejść dyskretnych, 16 punktowy	224 mikrosekund	68 mikrosekund
Moduł wejść dyskretnych, 16 punktowy	211 mikrosekund	47 mikrosekund
Moduł wejść dyskretnych, 32 punktowy	247 mikrosekund	91 mikrosekund
Moduł wyjść dyskretnych, 32 punktowy	226 mikrosekund	47 mikrosekund

SER

Charakterystyka

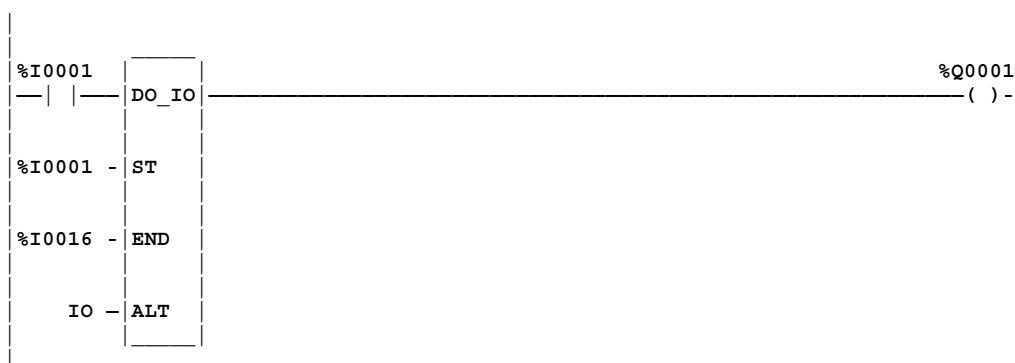
- Blok funkcyjny SER (Sequential Event Recorder) rejestruje serię próbek. Po doprowadzeniu sygnału do wejścia Enable, blok funkcyjny SER odczytuje próbkę, składającą się z maksymalnie 32 bitów, umieszczonych kolejno po sobie lub nie sąsiadujących.
- Każdy blok funkcyjny SER może zarejestrować do 1024 próbek.
- Jeżeli blok funkcyjny SER zostanie umieszczony w podprogramie wywoływanym okresowo, częstotliwość wykonywania zależy od częstotliwości wywoływania tego podprogramu.
- Czas jest zapisywany wyłącznie dla próbki rejestrowanej w momencie wyzwolenia (próbka rejestrowana). Czas dla próbki rejestrowanej może być zapisywany w formacie BCD (maksymalna częstotliwość wykonywania 1 s) lub w formacie POSIX (maksymalna częstotliwość wykonywania 10 ms). Czas jest zapisywany wyłącznie dla próbki zarejestrowanej w momencie doprowadzenia sygnału wyzwolenia. Blok funkcyjny SER nie rejestruje czasów dla innych próbek.
- Blok funkcyjny SER może pracować w następujących trybach: Rejestrowanie próbek do sygnału wyzwolenia, Rejestrowanie próbek do zebrania ich określonej liczby po sygnale wyzwolenia lub w trybie Rejestrowanie do zebrania określonej pozostałych próbek. (Proszę porównać z informacjami na stronie 12-14).
- Przebieg pracy bloku funkcyjnego SER zależy od zawartości bloku danych sterujących, który można utworzyć za pomocą funkcji Block Move (BLKMOV). (Proszę porównać z informacjami na stronie 12-10).

Uwaga

Nie jest obsługiwana synchronizacja pomiędzy sterownikami korzystającymi z bloków funkcyjnych SER.

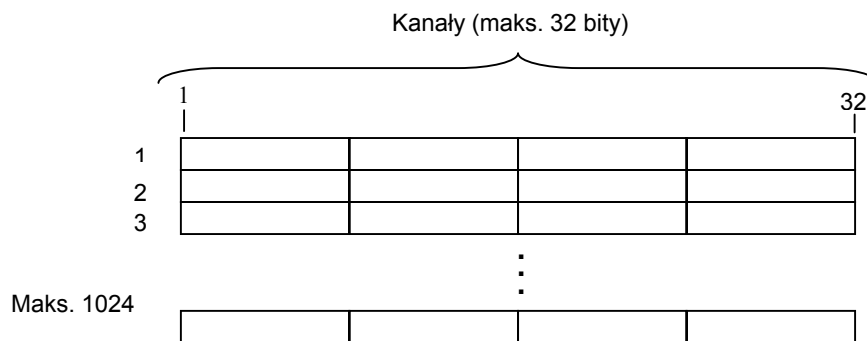
Blok funkcyjny COMMREQ posiada jeden parametr wyjściowy i trzy parametry wejściowe: enable, reset i trigger.

Przykład zastosowania bloku funkcyjnego SER



Uwaga

Blok funkcyjny SER wymaga oprogramowania systemowego jednostki centralnej wersja 9.00 i jest dostępny wyłącznie w jednostkach centralnych 350 i wyższych.



Parametry

Parametr	Opis
enable	Po doprowadzeniu sygnału wejściowego, jeżeli nie jest doprowadzany sygnał zerowania, blok funkcyjny SER odczytuje jedną próbkę ze wszystkich skonfigurowanych kanałów.
R	Sygnał zerowania. Po jego doprowadzeniu, blok funkcyjny SER jest zerowany, bez względu na wartość sygnału wejściowego. Powoduje to wyzerowanie parametrów Sample Buffer, Trigger Sample Offset, Trigger time i Current Sample Offset. Blok funkcyjny będzie pozostawał w stanie zerowania, aż do momentu kiedy sygnał zerowania przestanie być doprowadzany. W stanie zerowania, na wyjście OK nie jest wysyłany sygnał. Zaprzestanie doprowadzania sygnału do wejścia R powoduje wznowienie próbkowania.
T	Jeżeli wybrany został tryb Trigger Input Mode i blok funkcyjny SER jest aktywny, doprowadzenie sygnału na wejście T powoduje przełączenie bloku SER do trybu wyzwolenia. W trybie tym zapisywana jest próbka (tzw. próbka rejestrowana), indeks tej próbki oraz czas jej pobrania. W momencie wyzwolenia próbka zostaje zarejestrowana bez względu na liczbę pobranych do tej pory próbek. Po wyzwoleniu, rejestrator nadal kontynuuje pobieranie próbek, aż do momentu pobrania określonej liczby (określonej parametrem Number of Samples After Trigger), kiedy to następuje zatrzymanie pobierania próbek, do momentu doprowadzenia sygnału na wejście R. Sygnał wyzwolenia jest ignorowany po wybraniu trybu Full Buffer. Szczegółowe informacje odnośnie konfigurowania parametru Trigger Mode podano w punkcie „Blok danych sterujących”, na stronie 12-10.
Adres	Adres początkowy bloku danych sterujących, składającego się z 78 rejestrów. Blok danych sterujących zawiera parametry określające sposób wykonywania bloku funkcyjnego SER. Szczegółowe informacje podano w punkcie „Blok danych sterujących”, na stronie 12-9.
ok	Sygnał wyjściowy wysyłany po spełnieniu wszystkich warunków do wyzwolenia (określonych przez parametr Trigger Mode) oraz po zebraniu wszystkich próbek. Sygnał ten jest wysyłany bez względu na stan sygnału wejściowego, aż do momentu doprowadzenia sygnału na wejście zerujące.

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
Control Block								•				
R	•											
T	•											
ok	•											•

- Dozwolony typ parametru.

Blok danych sterujących

Blok danych sterujących jest tablicą złożoną z 78 elementów typu Word, definiującą sposób zbierania próbek oraz działanie mechanizmu wyzwalania. Każdy blok danych i blok danych sterujących może być wykorzystywany wyłącznie przez jeden blok funkcyjny SER.

W celu skonfigurowania parametrów bloku funkcyjnego SER należy postępować zgodnie z zamieszczoną poniżej procedurą:

1. Ustawić wartość elementów tablicy, zgodnie z zamieszczoną poniżej tabelą. Wartości rejestrów można zainicjować za pomocą funkcji kopiowania bloków pamięci lub zainicjować wartości w tabeli rejestrów, a następnie zapamiętać tę tabelę, przed wywołaniem bloku funkcyjnego SER.
2. Wprowadzić blok funkcyjny SER do szczebla programu sterującego.

Uwaga

Jeżeli użytkownik chce próbować x kanałów, gdzie x jest różne od 8, 16, 24 ale mniejsze od 32, musi wybrać liczbę kanałów większą od x i będącą wielokrotnością 8, a dla pozostałych, niewykorzystywanych kanałów należy wprowadzić pusty opis kanałów. Dla pustego opisu kanałów, parametr Segment Selector ma wartość 0xFFh, parametr długość musi być równy licznie niewykorzystywanych kanałów, a indeks wynosi 0.

Słowo	Parametr	Opis
0 (adres)	Status (<i>Status</i>)	Zmienna przeznaczona wyłącznie do odczytu, której zadaniem jest informowanie o bieżącym statusie bloku funkcyjnego SER. Dodatkowe informacje przekazywane są za pomocą słowa Status Extra Data (Słowo 1). Uwaga: Jeżeli w bloku danych sterujących wykryty zostanie błąd, wartość parametru Status jest ustawiana na 6, na wyjście OK przestanie być wysyłany sygnał i nie są realizowane żadne działania. Parametr status może przyjmować między innymi następujące wartości: 0 = Zerowanie 1 = Blok funkcyjny nieaktywny 2 = Blok funkcyjny aktywny 3 = Wyzwolenie 4 = Zakończenie działania 5 = Brak pamięci 6 = Błędny parametr
1	Status Extra Data (<i>Dodatkowe dane o statusie</i>)	Zmienna przeznaczona wyłącznie do odczytu, udostępniająca dodatkowe informacje o statusie bloku funkcyjnego SER. Szczegółowe informacje odnośnie tego parametru podano w punkcie „Słowo statusu Extra Data” na stronie 12-12.
2	Trigger mode (<i>Tryb wyzwalania</i>)	Parametr ten definiuje warunki, których spełnienie jest wymagane, aby blok funkcyjny SER przeszedł do stanu wyzwolenia. Parametr ten może przyjmować następujące wartości: 0 = Trigger Input mode (Wyzwalanie po doprowadzeniu sygnału do wejścia T) 1 = Full Buffer mode (Rejestrowanie do wypełnienia bufora) W trybie Trigger Input Mode, jeżeli blok funkcyjny SER jest aktywny, doprowadzenie sygnału na wejście T powoduje zapisanie czasu dla próbki rejestrowanej. Próbkowanie jest kontynuowane do momentu zebrania określonej liczby próbek (parametr Number of Samples After Trigger). Po zebraniu odpowiedniej liczby próbek, określonej tym parametrem, następuje wysłanie sygnału. W trybie rejestrowania Full Buffer ignorowane jest doprowadzenie sygnału do wejścia T. Jeżeli blok funkcyjny SER jest aktywny, pobieranie próbek jest kontynuowane do momentu wypełnienia bufora próbek. Po zebraniu odpowiedniej liczby próbek, określonej tym parametrem, następuje wysłanie sygnału wyjściowego OK. Rozmiar bufora jest określany za pomocą parametru Number of Samples.
3	Trigger Time Format (<i>Format czasu rejestrowania</i>)	Format wyświetlania czasu rejestrowania, w którym nastąpiło wyzwolenie bloku funkcyjnego SER. W przypadku wyświetlacza BCD, parametr ten należy ustawić na 0, natomiast dla wyświetlacza POSIX, parametr ten należy ustawić na 1 (szczegółowe informacje podano na stronie 12-20).

Słowo	Parametr	Opis
4—7	Zarezerwowany	Słowa 4 do 7 są słowami zarezerwowanymi, ich wartość powinna być ustawiona na zero (0).
8	Number of Channels (<i>Liczba kanałów</i>) (<i>bitów na próbkę</i>)	Liczba próbkowanych bitów danych, zwracanych do bufora próbek, po każdym wykonaniu bloku funkcyjnego SER. Dopuszczalnymi wartościami są 8, 16, 24 i 32 bity. Wartość tego parametru musi być wielokrotnością 8 bitów. (Proszę porównać ze słowami 14—77.)
9	Number of Samples (<i>Liczba próbek</i>)	Rozmiar bufora próbek w bajtach. Zakres dopuszczalnych wartości wynosi od 1 do 1024 próbek. (Rzeczywista wielkość bufora w bitach jest równa (Number of Samples)*(Number of Channels).
10	Number of Samples After Trigger (<i>Liczba próbek rejestrowanych po sygnale wyzwolenia</i>)	Liczba próbek zapamiętywanych w buforze próbek po wyzwoleniu bloku funkcyjnego SER. Parametr ten musi mieć wartość z zakresu od 0 do (Number of Samples -1). Jest on wykorzystywany wyłącznie jeżeli parametr Trigger Mode ma wartość Trigger Input (0).
11	Input Module Slot (<i>Gniazdo modułu wejść</i>)	Położenie modułu wejść, z którego odczytywane są próbki (gniazdo w kasecie głównej). Wprowadzenie wartości 0 powoduje wyłączenie odczytywania modułu wejść. Wartości odczytane z modułu wejść są zapamiętywane lokalnie, bez zmiany wartości zmiennych przypisanych do tego modułu. W celu zapisania wartości odczytanych z modułu wejść do bufora próbek w bloku danych należy podać opis kanału. Jeżeli moduł jest nie zainstalowany lub uszkodzony w momencie czytania, wartości te zostaną ustawione na zero. W przypadku wystąpienia takiego błędu, nie zostanie on zarejestrowany w tabeli błędów działania, jego zasygnalizowanie zostanie pozostawione dla skanera wejść i wyjść.
12	Data Block Segment Selector (<i>Typ pamięci bloku danych</i>)	Typ danych przypisanych przez użytkownika dla Bloku danych. Przykładowo, jeżeli blok danych ma rozpoczynać się pod adresem %R0100, należy wprowadzić wartość 08. Poprawnymi wartościami tego parametru są: %R (08h), %AI (0Ah), %AQ (0Ch). Szczegółowe informacje o bloku danych podano na stronie 12-13.
13	Data Block Offset (<i>Adres względny bloku danych</i>)	Adres początkowy bloku danych. Parametr ten jest liczony od zera. Przykładowo, jeżeli blok danych ma rozpoczynać się pod adresem %R0100, należy wprowadzić wartość 99. Użytkownik ponosi odpowiedzialność za przypisanie odpowiednio dużego obszaru pamięci dla całego bloku danych.
14—77	Channel Descriptions (<i>Opisy kanałów</i>)	Położenie zmiennej (Segment Selector, Length i Offset) związanej z danym kanałem. Można zgromadzić od 1 do 32 opisów kanałów, w zależności od liczby próbkowanych kanałów oraz od długości danych. Dane zwracane są w kolejności, zdefiniowanej w następnym punkcie.
	Channel Segment Selector/Length (<i>Typ pamięci kanału/ długość</i>)	Wartość ta wprowadzana jest w systemie heksadecymalnym, słowo to określa zarówno wybrany przez użytkownika typ danych jak i długość danych (w bitach). MSB = Wybór segmentu. LSB = Długość danych. Długość danych jest szczególnie użyteczna przy odczycie próbek z sąsiadujących ze sobą punktów. Parametr Segment Selector może wskazywać na dowolny obszar pamięci typu dyskretnego: %I (46h), %Q (48h), %M (4Ch), %T (4Ah), %G (56h), %S (54h), %SA (4Eh), %SB (50h), %SC (52h), typ pusty (FFh) i wybrany moduł wejściowy Module Selector(00h).
	Channel Offset (<i>Adres względny kanału</i>)	Wartość tego parametru wprowadzana jest w systemie heksadecymalnym, słowo to określa adres względny w bitach dla typu danych lub modułu wejść określonych przez podanie typu pamięci. Adres względny jest liczony od zera. Zakres wartości tego parametru zależy od parametru Segment Selector (typ danych i długość). Przesunięcie określa położenie w tabeli danych lub w module wejściowym, z którego pobierane są próbki.

Słowo Status Extra Data

Słowo Status Extra Data (Słowo 1 w bloku danych sterujących) zawiera dodatkowe informacje odnośnie statusu (słowo 0) bloku funkcyjnego SER.

Wartość	Status	Opis
0	Reset State	Doprowadzenie sygnału na wejście zerowania R powoduje wyzerowanie parametrów Sample Buffer, Trigger Sample Offset, Trigger time i Current Sample Offset. Nie jest wysyłany sygnał na wyjście logiczne. Jeżeli sygnał zerowania przestanie być doprowadzany, blok funkcyjny przejdzie do stanu <i>Inactive State</i> . Parametr Status Extra Data nie ma w takiej sytuacji żadnego znaczenia i jego wartość zostaje ustawiony na zero.
1	Inactive	Stan pomiędzy stanem zerowania (Reset State) a stanem aktywnego bloku funkcyjnego (Active State). W stanie tym nie są realizowane żadne działania. Blok funkcyjny SER nie wysyła sygnału wyjściowego. Po doprowadzeniu sygnału wejściowego, status zmieniany jest na Active State.
2	Active	Stan po doprowadzeniu sygnału wejściowego, jeżeli blok funkcyjny SER nie jest wyzerowany, nie stwierdzono błędów oraz nie nastąpiło wyzwolenie. W stanie tym, każde wykonanie tego bloku funkcyjnego powoduje zarejestrowanie jednej próbki. Nie jest wysyłany sygnał na wyjście logiczne. Monitorowane są warunki wymagane do wyzwolenia (określone za pomocą parametru Trigger Mode), a w przypadku ich spełnienia blok funkcyjny przechodzi w stan wyzwolenia (Triggered State). Jeżeli liczba pobranych próbek jest większa od liczby określonej za pomocą parametru Number of Samples, wartość parametru Status Extra Data jest równa 0x01, w przeciwnym wypadku jest równa 0x00.
3	Triggered	Status przyjmowany po spełnieniu warunków określonych parametrem Trigger Mode. Pobieranie dodatkowych próbek zależy od trybu wyzwolenia oraz od wartości parametrów. Nie jest wysyłany sygnał na wyjście logiczne. Po pobraniu wszystkich próbek, blok funkcyjny przechodzi w stan zakończenia działania (Complete State). Jeżeli liczba pobranych próbek jest większa od liczby określonej za pomocą parametru Number of Samples, wartość parametru Status Extra Data jest równa 0x01, w przeciwnym wypadku jest równa 0x00.
4	Complete	Blok funkcyjny SER przechodzi do tego stanu po zakończeniu pobierania wszystkich próbek. Wysyłany jest sygnał wyjściowy. Dopuszczalne jest przejście wyłącznie do stanu zerowania. Jeżeli liczba pobranych próbek jest większa od liczby określonej za pomocą parametru Number of Samples, wartość parametru Status Extra Data jest równa 0x01, w przeciwnym wypadku jest równa 0x00.
5	Overrun Error	Blok funkcyjny SER przechodzi do tego stanu, jeżeli Blok danych/ Blok danych sterujących przekroczył zakres przewidzianego dla niego typu pamięci. Nie jest wysyłany sygnał na wyjście logiczne. Dopuszczalne jest przejście wyłącznie do stanu zerowania. Parametr Status Extra Data nie ma w takiej sytuacji żadnego znaczenia i jego wartość zostaje ustawiony na zero.
6	Parameter Error	Wykrycie błędów w bloku danych sterujących lub w innych parametrach bloku funkcyjnego SER. Nie jest wysyłany sygnał na wyjście logiczne. Dopuszczalne jest przejście wyłącznie do stanu zerowania. Parametr Status Extra Data zawiera pozycję w bloku danych sterujących, określając tym samym położenie błędnego parametru.
7	Status Error	Nieprawidłowa wartość parametru Status. Nie jest wysyłany sygnał na wyjście logiczne. Dopuszczalne jest przejście wyłącznie do stanu zerowania. Nieprawidłowa wartość jest zapisywana w Bloku sterującym w miejscu określonym parametrem Status Extra Data.

Blok danych

Blok danych SER zawiera bufor próbek, indeksy próbek oraz informacje o wyzwoleniu. Informacje te są zapisywane przez jednostkę centralną, są one przeznaczone wyłącznie do odczytu. Użytkownik jest odpowiedzialny za przypisanie dostatecznie dużego obszaru pamięci dla bloku danych. Blok ten ma następującą strukturę:

Słowo*	Opis parametru
0	Numer indeksu dla bieżącej próbki. Parametr ten podaje indeks ostatnio zarejestrowanej próbki. Indeks ten liczony jest od zera. Zakres dopuszczalnych wartości wynosi od -1 do 1023. Indeks rejestru zawierającego próbkę = (Liczba bajtów na próbkę) * (Indeks) + (Rejestr początkowy bufora próbek). Uwaga: Wartość ta zostaje określona dopiero w momencie wyzwolenia. W momencie zerowania bloku funkcyjnego SER (poprzez doprowadzenie sygnału na wejście R), zostaje ona ustawiona na -1.
1	Numer indeksu dla próbki rejestrowanej. Miejsce w którym zostanie zapisana próbka w momencie wyzwolenia. Indeks ten liczony jest od zera. Zakres dopuszczalnych wartości wynosi od 0 do 1023. Indeks rejestru zawierającego próbkę = (Liczba bajtów na próbkę) * (Indeks) + (Rejestr początkowy bufora próbek). Uwaga: Wartość ta zostaje określona dopiero w momencie wyzwolenia, w związku z czym nie należy z niej korzystać przed wyzwoleniem. W momencie zerowania bloku funkcyjnego SER za pośrednictwem sygnału zerującego, zostaje ona ustawiona na 0.
2 do 5	Czas wyzwolenia: Jest to czas wyzwolenia, zapisywany według wskazań zegara czasu rzeczywistego sterownika, w którym nastąpiło spełnienie warunków wyzwolenia. Wartość czasu wyświetlana jest (domyślnie) w formacie BCD, ale istnieje również możliwość wyświetlania w formacie POSIX. Format wyświetlania określany jest w bloku danych sterujących za pomocą parametru <i>Trigger Time Format</i> . Doprowadzenie sygnału zerującego powoduje ustawienie tej wartości na zero.
6 do końca bufora.	Bufor próbek. Obszar w pamięci, w którym przechowywane są próbki danych. Obszar ten jest zerowany po doprowadzeniu sygnału zerowania. Rozmiar bufora próbek zależy od liczby kanałów oraz wielkości jednej próbki. Jest to bufor o kołowej strukturze - po zapisaniu na ostatniej pozycji, następna próbka zostanie zapisana w pierwszym rejestrze. Adres końca bufora próbek = $5 + ((\text{liczba próbek do pobrania}) * (\text{liczba kanałów do próbkowania} / 8)) + 1 / 2$

*Kolumna 1 podaje adres względny liczony od adresu początkowego, zdefiniowanego za pomocą parametrów Data Block Segment Selector (Słowo 12) i Data Block Offset (Słowo 13) w bloku danych sterujących.

Opis działania

Odczytywane próbki są umieszczane w tablicy o strukturze kołowej. Sygnał wyjściowy jest wysyłany po odczytaniu skonfigurowanej liczby próbek. Sygnał wyjściowy można wykorzystać w celu odnotowania czasu pobrania ostatniej próbki lub w celu rozpoczęcia zbierania dodatkowych próbek.

Blok funkcyjny SER musi zostać wyzerowany (poprzez doprowadzenie sygnału zerowania) przed rozpoczęciem próbkowania. Wyzerowanie powoduje zainicjowanie obszaru pamięci dla bloku danych. Jeżeli blok funkcyjny SER nie zostanie wyzerowany, będzie nadal korzystał z tego samego bloku danych, na skutek czego indeks bieżącej próbki oraz dane zapisane w bloku danych będą niepoprawne.

Blok danych sterujących jest odczytywany przy każdym wykonywaniu tego bloku funkcyjnego, w stanie zerowania, wyzwolenia lub jeżeli blok funkcyjny jest aktywny. Jeżeli użytkownik zmieni jeden z parametrów konfiguracyjnych Bloku danych sterujących w czasie wykonywania programu, zmiana zostanie uwzględniona przy następnym wywołaniu bloku funkcyjnego SER, powiązanego z tym Blokiem danych sterujących. W przypadku wystąpienia błędu, praca zostanie zatrzymana a blok funkcyjny SER przechodzi do odpowiedniego stanu, w zależności od rodzaju napotkanego błędu. W celu wznowienia próbkowania, użytkownik musi usunąć przyczynę błędu i wyzerować blok funkcyjny (poprzez doprowadzenie sygnału do wejścia zerowania).

Jeżeli próbki odczytywane są z modułu wejść, sterownik nie sprawdza, czy jest to moduł wejść dyskretnych oraz czy opisy kanałów tego modułu mają długości i pozycje dostosowane do wielkości modułu. Użytkownik jest odpowiedzialny za wprowadzenie poprawnych parametrów próbkowania dla modułu wejść. Moduł wejściowy może mieć wiele opisów kanałów, ale przy każdym wykonaniu bloku funkcyjnego będzie on nadal obsługiwany wyłącznie jeden raz.

Blok funkcyjny SER może być wykorzystywany w normalnym bloku programu sterującego lub w okresowo wywoływanych podprogramach. W przypadku umieszczenia w normalnym bloku programu sterującego, częstotliwość wykonywania bloku funkcyjnego SER zależy od czasu trwania cyklu pracy sterownika, a więc od ilości i rodzaju aktywnych w danym cyklu bloków funkcyjnych. W przypadku umieszczenia go w podprogramie wywoływanym przez przerwanie, blok SER może być wywoływany nawet co 1ms, z bardzo niewielkimi odchyleniami od tej wartości.

Wykonywanie bloku funkcyjnego SER co 1ms może spowodować zajęcie prawie 50% zasobów jednostki centralnej. W podprogramie wykonywanym co 1ms nie należy umieszczać więcej niż dwóch bloków funkcyjnych SER.

Tryby próbkowania

Tryb próbkowania jest określany przez parametr Trigger Mode (Słowo 2 w bloku danych sterujących) i przez parametr Number of Samples After Trigger (Słowo 10). Interpretacja zawartości bufora próbek zależy od wartości tych parametrów.

Próbkowanie sterowane sygnałem wyzwania

W celu skonfigurowania trybu: Rejestrowanie próbek do sygnału wyzwolenia, Rejestrowanie próbek do zebrania ich określonej liczby po sygnale wyzwolenia lub w trybie Rejestrowanie do zebrania określonej liczby próbek należy parametr Trigger Mode (słowo 2) ustawić na 0. Należy dodatkowo skonfigurować parametr Number of Samples After Trigger (Słowo 10) w celu określenia trybu próbkowania. We wszystkich przypadkach próbkowanie jest rozpoczynane po doprowadzeniu sygnału wejściowego. Po doprowadzeniu sygnału na wejście wyzwania (T), próbkowanie jest kontynuowane do momentu zebrania odpowiedniej liczby próbek (określonej za pomocą parametru Number of Samples After Trigger). Po zakończeniu próbkowania następuje wysłanie sygnału wyjściowego.

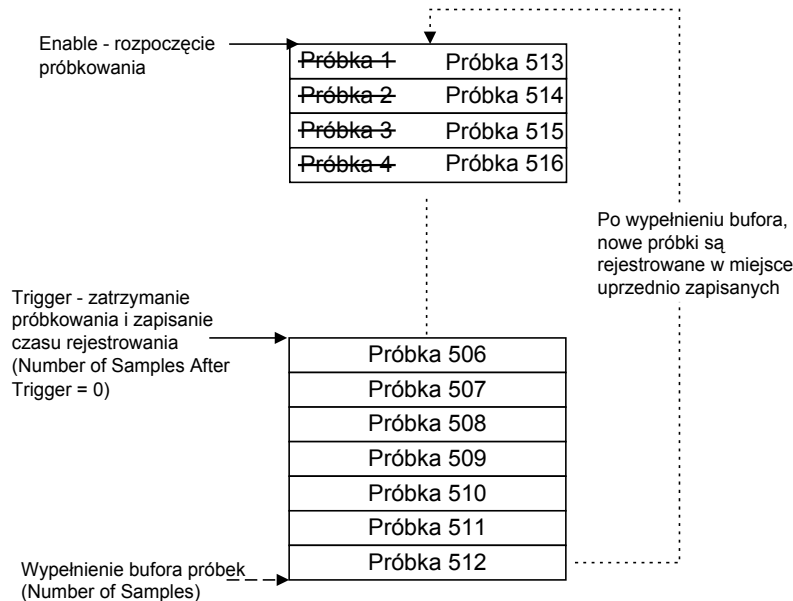
Jeżeli przed zebraniem liczby próbek określonej parametrem Number of Samples After Trigger, dotychczas zebrana liczba próbek jest większa od wartości parametru Number of Samples (Słowo 9), bufor zostanie "przewinięty", tzn. blok funkcyjny rozpoczyna zapisywanie na jego początku, w miejsce próbek zebranych wcześniej.

Zmiana stanu sygnału wyzwania z 0 na 1 powoduje zapisanie czasu wyzwolenia do określonego miejsca w bloku danych.

Tryb Rejestrowanie próbek do sygnału wyzwolenia

W trybie tym próbki są rejestrowane do momentu doprowadzenia sygnału wyzwolenia.

W celu skonfigurowania tego trybu, należy w Słowie 10 zapisać wartość 0, co spowoduje, że po doprowadzeniu sygnału wyzwolenia, próbkowanie zostanie zatrzymane i w określonym miejscu w bloku danych (słowa o indeksach 2-5) zapisany zostanie Czas rejestrowania. (Wszystkie próbki zebrane przed wyzwoleniem).

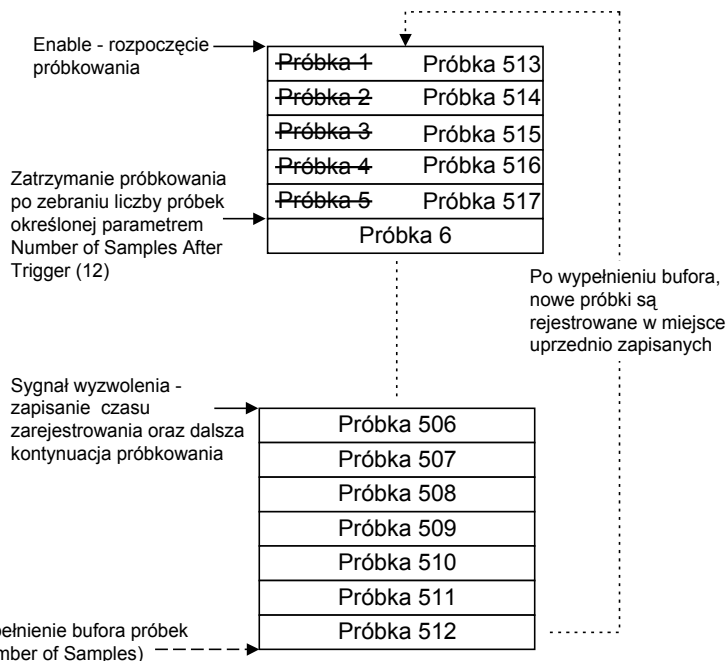


Rysunek 12-1. Przykład próbkowania w trybie Rejestrowanie próbek do sygnału wyzwolenia

Tryb Rejestrowanie próbek do zebrania ich zadanej liczby po doprowadzeniu sygnału wyzwolenia.

W trybie tym, po doprowadzeniu sygnału wyzwolenia, próbkowanie jest kontynuowane do momentu zebrania określonej liczby próbek, zadanej parametrem Number of Samples After Trigger (Słowo 10).

W celu skonfigurowania tego trybu, należy w Słowie 10 zapisać wartość z zakresu 1 do Number of Samples (słowo 9). Po doprowadzeniu sygnału na wejście wyzwalania (T), próbkowanie jest kontynuowane do momentu zebrania określonej liczby próbek. W zamieszczonym poniżej przykładzie, parametr Number of Samples After Trigger ustawiono na 12. Po zakończeniu próbkowania, bufor będzie zawierał 500 próbek zapisanych przed wyzwoleniem oraz 12 próbek zapisanych po wyzwoleniu.

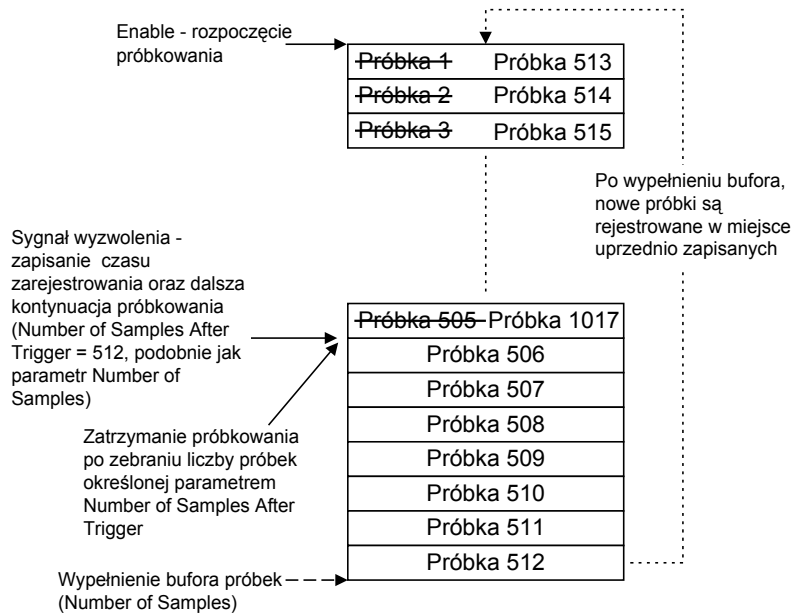


Rysunek 12-2. Przykład próbkowania w trybie Rejestrowanie próbek do zebrania ich odpowiednich liczby po doprowadzeniu sygnału wyzwolenia

Tryb Rejestrowanie do zebrania określonej liczby próbek

W trybie tym, próbkowanie jest kontynuowane przez cały czas do momentu zebrania liczby próbek określonej parametrem Number of Samples.

W celu skonfigurowania tego trybu, należy w Słowie 10 zapisać wartość równą wartości parametru Number of Samples (Word 9). Po doprowadzeniu sygnału na wejście wyzwiania (T), próbkowanie jest kontynuowane do momentu zebrania określonej liczby próbek. (Wszystkie zebrane po wyzwoleniu).



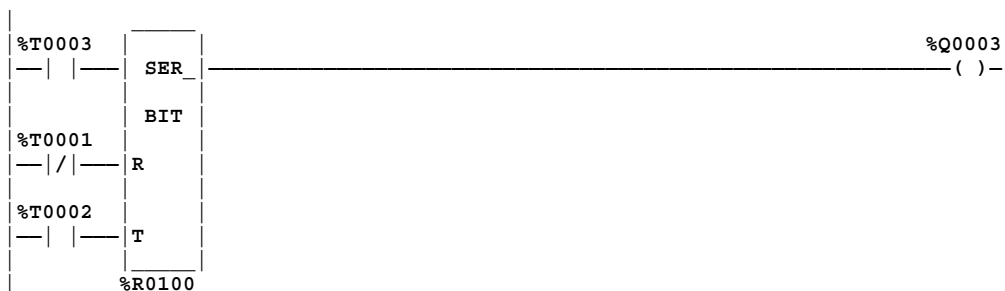
Rysunek 12-3. Przykład próbkowania w trybie rejestrowania do zebrania określonej liczby próbek

Tryb Rejestrowanie do wypełnienia bufora

Jeżeli parametr Trigger Mode zostanie ustawiony na 1, ignorowany jest parametr Number of Samples After Trigger (Słowo 10), a sygnał wyzwiania nie ma wpływu na sposób wykonywania bloku funkcyjnego SER. Jeżeli blok funkcyjny SER jest aktywny, pobieranie próbek jest kontynuowane do momentu zebrania liczby próbek określonej za pomocą parametru Number of Samples (Słowo 8). Po wypełnieniu bufora następuje zatrzymanie próbkowania, zapisanie czasu rejestrowania oraz wysłanie sygnału wyjściowego.

Przykłady

W zamieszczonym poniżej przykładzie wykorzystywany jest blok danych sterujących przedstawiony w Tabeli 12-1.



Przykład bloku danych sterujących

Na użytek niniejszego przykładu założone zostanie, że system wyposażony jest w moduł z 16 wejściami dyskretnymi, zainstalowany w kasecie 0, gnieździe 4, pracował przez dostatecznie długi czas, pozwalający na zarejestrowanie 572 próbek (512 + 60). Sygnał jest doprowadzany do wejścia Enable, a nie jest doprowadzany do wejść Reset (R) i Trigger (T).

Tabela 12-1. Blok danych sterujących dla przykładowego bloku funkcyjnego SER

Słowo	Rejestr	Parametr	Wartość (dziesiąt.)	Wartość (heksadec.)	Opis
0	%R0100	Status	2	0002	Blok funkcyjny SER znajduje się w stanie aktywnym. Oznacza to, że blok funkcyjny jest normalnie wykonywany, a próbka pobierana jest za każdym razem, kiedy w programie sterującym zostanie napotkany ten blok funkcyjny.
1	101	Status Extra Data	1	0001	Słowo Extra Status Data sygnalizuje, że pobrano więcej niż 512 próbek, a więc nastąpiło co najmniej jedno przesunięcie w buforze próbek o strukturze kołowej.
2	102	Trigger mode	0	0000	Blok funkcyjny SER przejdzie do stanu wyzwolenia po doprowadzeniu sygnału do wejścia T.
3	103	Trigger Time Format	0	0000	BCD
4	104	Zarezerwowany	0	0000	Parametry zarezerwowane mają zawsze wartość 0.
5	105	Zarezerwowany	0	0000	
6	106	Zarezerwowany	0	0000	
7	107	Zarezerwowany	0	0000	
8	108	No of channels	24	0018	Każda próbka zawiera 24 bity danych.
9	109	No of samples to be taken	512	0200	Wielkość bufora próbek wynosi 512. Należy zwrócić uwagę, że bufor próbek nie zajmuje 512 bajtów. Jego wielkość wynosi $512 \times (24/8) = 1536$ bajtów lub 768 słów. (Każda z próbek ma 3 bajty długości).
10	110	No of samples after trigger	12	000C	Po każdym wyzwoleniu zapisywanych jest 12 próbek.
11	111	Input module slot	4	0004	Dane będą czytane z modułu wejść zainstalowanego w kasecie 0, gnieździe 4.
12	112	Data Block Segment Selector	8	0008	Blok danych będzie zapisany w pamięci %R (0x08).
13	113	Data Block Offset	200	00C8	Adres względny wynosi 200, a więc adres początkowy bloku danych to %R0201. Adres względny jest liczony od zera, ale tablica rejestrów rozpoczyna się od %R0001. Z tego powodu, punkt początkowy bloku danych jest równy $\%R0001 + 200 = \%R0201$.

Słowo	Rejestr	Parametr	Wartość (dziesiętn.)	Wartość (heksadec.)	Opis
Channel Descriptions		Pozostałe słowa zawierają opisy kanałów. W niniejszym przykładzie zdefiniowano 6 opisów kanałów.			
14	114	Seg. Sel. : Length	17921	4601	Opis kanału 1: dla pierwszego kanału wybrano Segment %I o długości 1 i adresie względnym 0. Dla kanału 1 wybrana została więc zmienna %I0001.
15	115	Poz.	0	0000	
16	116	Seg. Sel. : Length	-253	FF03	Opis kanału 2: Dla drugiego kanału wybrano segment zerowy o długości 3 i adresie względnym 0. Powoduje to pominięcie ("zignorowanie") kanałów 2 - 4. Wartość próbki dla tych kanałów będzie zawsze równa zero.
17	117	Poz.	0	0000	
18	118	Seg. Sel. : Length	3	0003	Opis kanału 3: Wybrano moduł wejść o długości 3 i adresie względnym 12. Powoduje to bezpośrednie pobieranie próbek z modułu wejść. W tym opisie kanału wybrano dla kanałów 5 - 7 wartości punktów 13, 14 i 15 modułu wejść.
19	119	Poz.	12	0012	
20	120	Seg. Sel. : Length	18434	4802	Opis kanału 4: W opisie czwartego kanału wybrano Segment %Q o długości 2 i adresie względnym 8. Tak więc kanałom 8 i 9 przypisane są %Q0009 i %Q0010.
21	121	Poz.	8	0008	
22	122	Seg. Sel. : Length	8	0008	Opis kanału 5: dla piątego kanału wybrano ponownie moduł wejść. Posiada on długość 8 i adresie względnym 0. Spowoduje to, że wartości punktów 1 do 8 modułu wejść będą umieszczane w kanałach 10 - 17.
23	123	Poz.	0	0000	
24	124	Seg. Sel. : Length	-249	FF07	Opis kanału 6: dla szóstego kanału wybrano opis pusty. Posiada on długość 7 i adresie względnym 0. Ten zerowy opis kanału powoduje że kanały 18 - 24 będą wypełniane zerami. Ten ostatni opis kanału wymagany jest do uzupełnienia bufora próbek do 24 bitów. Ponieważ skonfigurowano tutaj wszystkie 24 kanały, nie są potrzebne jakiegokolwiek inne opisy kanałów.
25	125	Poz.	0	0000	

Zawartość próbki

W zamieszczonej poniżej tabeli zestawiono wartości zapisane dla pojedynczej próbki, w oparciu o opisy kanałów zamieszczone w bloku danych sterujących.

Tabela 12-2. Zawartość próbki dla przykładowego bloku funkcyjnego SER

Numer kanału	Zawartość kanału
1	%I0001
2 - 4	Zera
5	Punkt 13 modułu wejść
6	Punkt 14 modułu wejść
7	Punkt 15 modułu wejść
8	%Q0009
9	%Q0010
10 - 17	Punkty 1 - 8 modułu wejść
18 - 24	Zera

Blok danych dla przykładowego bloku danych sterujących

Zamieszczona poniżej tabela podaje format bloku danych, dla przedstawionego wcześniej, przykładowego bloku danych sterujących. Należy zwrócić uwagę, że blok ten rozpoczyna się w rejestrze 201, zgodnie z parametrami wprowadzonymi w bloku sterującym (Słowo 12 i Słowo 13).

Tabela 12-3. Blok danych dla przykładowego bloku funkcyjnego SER

Adres względny	Rejestr	Opis parametru	Wartość (dziesiętna)	Wartość (heksadecym.)
0	%R0201	Adres względny dla bieżącej próbki:	59	003B
1	202	Adres względny dla rejestrowanej próbki:	0	0000
2 - 5	203 – 206	Czas zarejestrowania (BCD)	0 0 0 0	0000 0000 0000 0000
6 - 768	207 – 975	Bufor próbek.	dane dla próbki	dane dla próbki

Numer indeksu bieżącej próbki wynosi 59, co oznacza, że ostatnią z zapisanych w buforze próbek jest próbka o numerze 59 (nie 59 rejestrów). Ponieważ każda próbka zajmuje 3 bajty, indeks wynosi $59 \cdot 3 = 177$ bajtów lub inaczej, jest to starszy bajt 89-o rejestru. Ponieważ w tym momencie nie są spełnione warunki do wyzwolenia, zarejestrowana próbka i czas wyzwolenia są równe 0 oraz nie jest wysyłany sygnał wyjściowy. W buforze zapisanych jest 512 próbek, ostatnio odczytana próbka zajmuje pozycję 59, a najstarsza pozycję 60.

Przykład formatów zapisu czasu rejestrowania

Czas rejestrowania: 3 listopad, 1998, godz. 8:34:05:16.

BCD Format:

```
struct time_of_day_clk_rec {
    unsigned char  seconds;
    unsigned char  minutes;
    unsigned char  hours;
    unsigned char  day_of_month;
    unsigned char  month;
    unsigned char  year;
};
```

Rejestr	Parametr	Wartość (dziesiętna)	Wartość (heksadecym.)
%R0203	Minuty/ sekundy	13317	3405
%R204	Dzień miesiąca/ godzina	776	0308
%R205	Rok/ miesiąc	-26607	9811
%R206	Nie wykorzystywany	0	0

Format POSIX:

```
struct timespec {
    long    tv_sec; /* Liczba sekund od 1 stycznia 1970 */
    long    tv_nsec; /* Liczba nanosekund do następnej sekundy */
};
```

Rejestr	Parametr	Wartość (dziesiętna)	Wartość (heksadecym.)
%R0203	Młodsze słowo liczby sekund	-7811	e17d
%R204	Starsze słowo liczby sekund	13845	3615
%R205	Młodsze słowo liczby nanosekund	26624	6800
%R206	Starsze słowo liczby nanosekund	2441	0989

END

Funkcja END przerywa wykonanie części logicznej programu sterującego. Program wykonywany jest począwszy od pierwszego szczebla drabiny logicznej aż do szczebla ostatniego lub do momentu napotkania funkcji END.

Instrukcja END powoduje bezwarunkowe przerwanie wykonywania programu. Po funkcji END nie można w szczeblu umieszczać żadnych innych instrukcji. Część programu sterującego, umieszczona po funkcji END, nie jest dalej wykonywana, w następnym cyklu program jest wykonywany od początku.

Funkcja END może być wykorzystana w trakcie uruchamiania programu do blokowania wykonywania części programu sterującego, umieszczonego po tej instrukcji.

W oprogramowaniu Logicmaster 90 przewidziany jest element wskazujący koniec części logicznej programu sterującego: [END OF PROGRAM LOGIC]. Element ten jest wykorzystywany, jeżeli w programie sterującym nie umieszczono funkcji END.

- [END]

Przykład

W zamieszczonym poniżej przykładzie, instrukcja END powoduje zakończenie wykonywania programu sterującego w bieżącym cyklu.

```
STOP  
- [ END ]
```

Uwaga

Użycie funkcji END w programie sterującym SFC lub w programie wywoływanym z poziomu SFC powoduje wygenerowanie w jednostkach centralnych z oprogramowaniem systemowym wer. 7 i nowszych błędu "END Function Executed from SFC Action".

(W jednostkach centralnych z oprogramowaniem systemowym starszym od wer. 7, funkcja ta pracuje nie prawidłowo, ale nie jest generowany błąd). Więcej informacji na temat tego błędu podano w Rozdziale 3 w punkcie "System Configuration Mismatch".

MCRN/MCR

Funkcja MCR musi być stosowana wraz z odpowiadającą jej funkcją ENDMCR. Obydwie te funkcje muszą posiadać taką samą etykietę. Wszystkie szczeble programu sterującego, umieszczone pomiędzy funkcją MCR i odpowiadającą jej funkcją ENDMCR zostaną wykonane bez dopływu sygnału sterującego. Po napotkaniu funkcji ENDMCR wznawiane jest normalne wykonywanie programu sterującego. W odróżnieniu od instrukcji JUMP, instrukcja MCR może być użyta tylko w kierunku do przodu programu (nie wstecz). Po instrukcji MCR musi występować odpowiadająca jej instrukcja ENDMCR.

Instrukcja MCR zmienia sposób działania następujących elementów programu sterującego:

- Przełączniki czasowe nie są inkrementowane lub dekrementowane. Typy TMR są zerowane. W przypadku bloku funkcyjnego ONDTR, zachowywana jest zliczona wartość.
- Normalne wyjścia są zerowane; wyjścia zanegowane są ustawiane na 1.

Uwaga

Po doprowadzeniu sygnału do funkcji MCR, znajdujący się w jej zasięgu fragment programu sterującego jest wykonywany, wyświetlany jest stan styków, ale nie są wysyłane sygnały wyjściowe. Jeżeli użytkownik nie ma świadomości, że dany fragment programu sterującego znajduje się w zasięgu instrukcji MCR, może podejrzewać błędne wykonywanie programu sterującego. W celu zasygnalizowania, że dany szczebel programu sterującego znajduje się pod kontrolą funkcji MCR, oprogramowanie wyświetla na ekranie podwójną szynę sygnałów.

Oprogramowanie Logicmaster 90-30/20/Micro udostępnia dwa rodzaje funkcji MCR: bez możliwości zagnieżdżania (MCR) i z możliwością zagnieżdżania (MCRN).

Kompatybilność z jednostkami centralnymi

Typ jednostki centralnej	Porównanie z maskowaniem
Jednostki centralne 35x i 36x (wer. 2 i nowsze)	Dozwolone jest stosowanie wyłącznie formy zagnieżdżonej (MCRN)
Jednostki centralnej serii 90, wer. 1	Dozwolone jest stosowanie wyłącznie formy nie zagnieżdżonej (MCR)

Opis działania funkcji MCRN

Funkcja MCRN może być umieszczana w dowolnym miejscu w programie, pod warunkiem, że jest poprawnie zagnieżdżona wewnątrz innych funkcji MCRN oraz, że zakres jej działania nie pokrywa się z zakresem działania funkcji MCR i JUMP.

Funkcja MCRN i odpowiadająca jej funkcja ENDMCRN muszą być całkowicie zawarte wewnątrz innej pary funkcji MCRN/ ENDMCRN. Dopuszczalne jest zagnieżdżanie do 8 poziomów. Przykład podano na stronie 12-24.

Uwaga

W jednostkach centralnych serii 35x i 36x, każdej funkcji ENDMCRN musi odpowiadać dokładnie jedna (1) funkcja MCRN.

Jednej funkcji ENDMCRN może odpowiadać kilka funkcji MCRN (za wyjątkiem jednostek centralnych serii 35x i 36x, proszę porównać z zamieszczoną powyżej uwagą). Jest to właściwość analogiczna do tej, jaką posiadają instrukcje JUMP/LABEL z możliwością zagnieżdżania - jednej instrukcji LABEL może

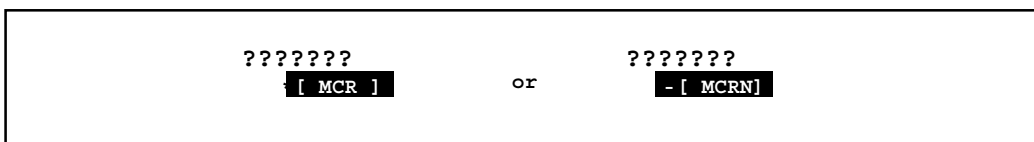
odpowiadać kilka instrukcji JUMP. Różnice pomiędzy funkcją JUMP i funkcją MCR omówiono w punkcie "Różnice pomiędzy funkcjami MCR i JUMP", na stronie 12-23.

Opis działania funkcji MCR

Na jedną funkcję MCR może przypadać tylko jedna funkcja ENDMCR. Zakres działania określony parami funkcji MCR/ ENDMCR nie może zawierać się wewnątrz zakresu działania innej pary funkcji MCR/ENDMCR ani też pary JUMP/LABEL. Funkcji MCR nie można umieszczać wewnątrz zakresu działania innej pary funkcji MCR/ ENDMCR lub pary funkcji JUMP/ LABEL.

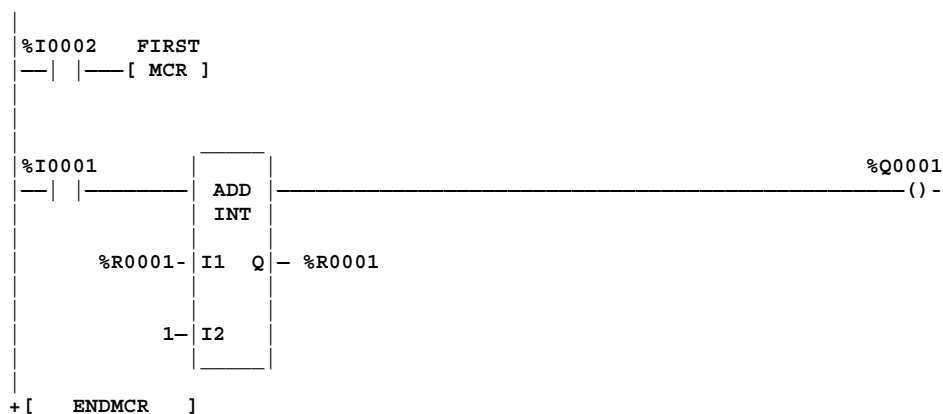
Parametry

Obydwie formy funkcji MCR posiadają takie same parametry. Są to: sygnał aktywujący EN typu bitowego oraz etykieta. Etykieta ta powinna być użyta przy odpowiedniej funkcji ENDMCR lub ENDMCRN. Funkcje MCR oraz MCRN nie posiadają parametrów wyjściowych. Muszą być one ostatnimi instrukcjami w szczeblu.

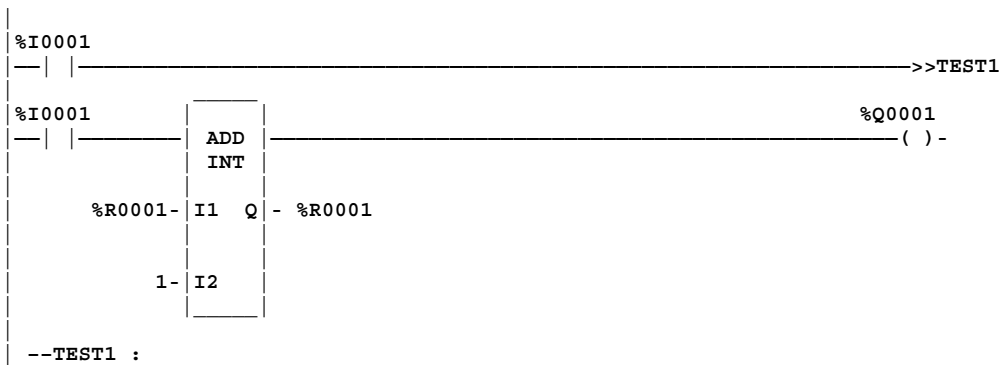


Różnice pomiędzy funkcjami MCR i JUMP

Funkcje znajdujące się w zakresie działania funkcji MCR są wykonywane *bez dopływu sygnału sterującego*, wartość zmiennych przypisanych do przekaźników *ustawiana jest na 0*. W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0002 na 1 powoduje aktywowanie funkcji MCR. Jeżeli aktywna jest funkcja MCR - nawet jeżeli wartość zmiennej %I0001 jest równa 1 – funkcja ADD wykonywana jest *bez* doprowadzania sygnału sterującego (tzn. nie dodaje on wartości 1 do zmiennej %R0001), a zmienna %Q0001 jest równa 0.



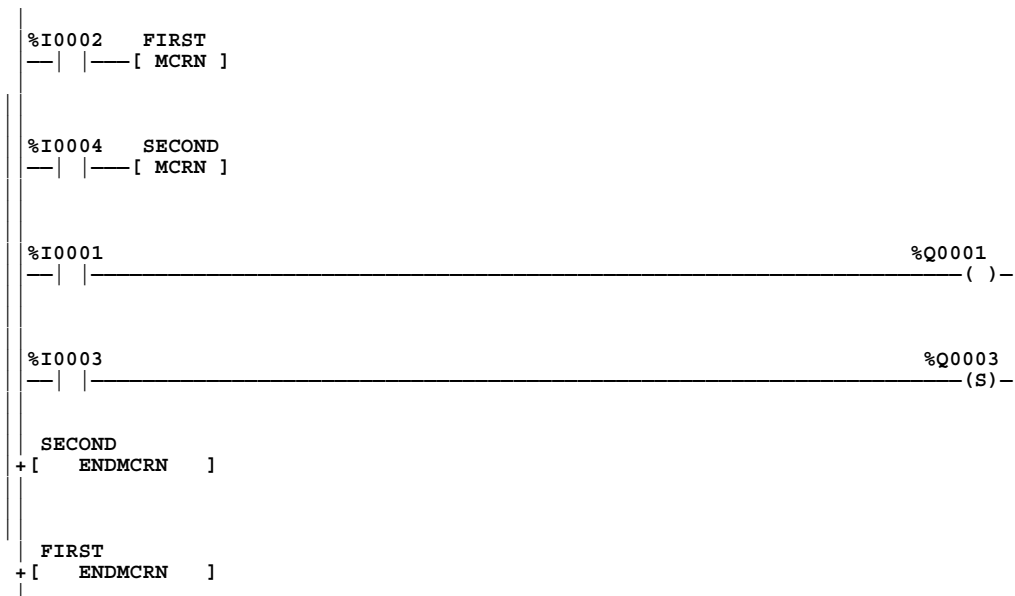
W przypadku funkcji JUMP, wszystkie funkcje pomiędzy instrukcjami JUMP i LABEL *nie są* wykonywane oraz *nie jest zmieniana* wartość zmiennych przekaźnikowych. W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0002 na 1 powoduje wykonanie instrukcji JUMP. Ponieważ funkcje umieszczone pomiędzy instrukcjami JUMP i LABEL zostają pominięte, wartość zmiennej %Q0001 nie ulega zmianie (tzn. jeżeli miała ona wartość 1, pozostawiana jest wartość 1, a jeżeli miała wartość 0, pozostawiana jest wartość 0).



Przykład

W zamieszczonym poniżej przykładzie pokazano instrukcję MCRN o nazwie "Second" zagnieżdżoną wewnątrz instrukcji MCRN o nazwie "First". Każda zmiana wartości zmiennej %I0002 na 1 powoduje doprowadzenie sygnału do instrukcji MCR i wykonywanie programu sterującego, bez doprowadzania sygnału sterującego do przekaźników, aż do momentu napotkania powiązanej z tą instrukcją instrukcji ENDMCR. Jeżeli zmienne %I0001 i %I0003 są równe 1, wartość zmiennej %Q0001 zostaje ustawiona na 0, a wartość zmiennej %Q0003, równa 1, nie ulega zmianie.

Część programu znajdującą się pod kontrolą MCR jest zaznaczana poprzez wyświetlenie podwójnej szyny sygnałów.

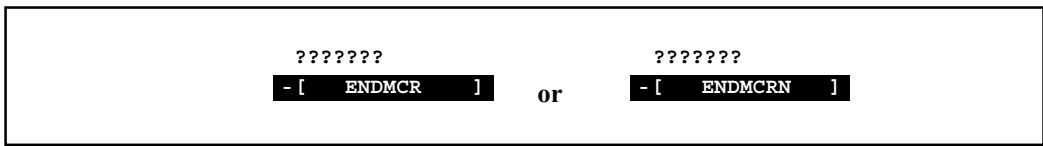


ENDMCRN/ENDMCR

Funkcja ENDMCR przywraca normalne wykonywanie programu sterującego (z normalnym dopływem sygnału do szczebli programu). Jeżeli do funkcji MCR dopływa sygnał wejściowy, ENDMCR kończy działanie tej funkcji. Jeżeli do funkcji MCR nie dopływa sygnał wejściowy, wywołanie funkcji ENDMCR nie powoduje żadnego działania.

Oprogramowanie Logimaster 90-30/20/Micro udostępnia dwa rodzaje funkcji MCR: bez możliwości zagnieżdżenia (MCR) i z możliwością zagnieżdżenia (MCRN). Instrukcja ENDMCR może być stosowana wyłącznie dla instrukcji MCR (a nie MCRN). Instrukcja ENDMCRN może być natomiast stosowana wyłącznie dla instrukcji MCRN.

Pierwszym parametrem funkcji ENDMCR jest zanegowane wejście sygnału sterującego EN. Sygnał wejściowy musi być doprowadzany bezpośrednio z szyny sygnału, nie jest dopuszczalne stosowanie warunkowego wykonywania. Drugim parametrem jest etykieta, wiążąca ją z odpowiednią funkcją (funkcjami) MCR. Funkcje ENDMCR oraz ENDMCRN nie posiadają parametrów wyjściowych. Powinny one być jedynymi instrukcjami w swoich szczeblach.



Przykład

W zamieszczonym poniżej przykładzie, funkcja ENDMCR powoduje przerwanie działania funkcji MCR "clear".

Przykład nie zagnieżdżonej funkcji ENDMCR:

```

CLEAR
- [ ENDMCR ]

```

Przykład zagnieżdżonej instrukcji ENDMCR:

```

CLEAR
- [ ENDMCRN ]

```

JUMP

Funkcja JUMP powoduje pominięcie fragmentu części logicznej programu sterującego, zamieszczonego pomiędzy instrukcją JUMP a instrukcją LABEL (etykieta). Po doprowadzeniu do funkcji sygnału wejściowego JUMP, wszystkie przełączniki zawarte we wspomnianym obszarze zachowują swój pierwotny stan (dotyczy to również takich elementów logicznych, jak przełączniki czasowe, liczniki, itp.).

Oprogramowanie Logicmaster 90 posiada dwa typy instrukcji skoku: bez możliwości zagnieżdżania i z możliwością zagnieżdżania. Pierwsza z nich, bez możliwości zagnieżdżania, dostępna począwszy o wersji 1 oprogramowania systemowego, oznaczana jest symbolem, `—————>>LABEL01`, gdzie LABEL01 jest nazwą nie zagnieżdżonej instrukcji LABEL.

W przypadku instrukcji JUMP bez możliwości zagnieżdżania, każdej instrukcji JUMP odpowiada dokładnie jedna instrukcja LABEL. Instrukcja JUMP może powodować przejście zarówno w kierunku do przodu, jak i do tyłu programu sterującego.

Zakres działania określony parami instrukcji MCR/ ENDMCR nie może zawierać się wewnątrz zakresu działania innej pary instrukcji MCR/ENDMCR ani też pary JUMP/LABEL. Instrukcje JUMP bez możliwości zagnieżdżania nie mogą być umieszczane wewnątrz zakresu działania innej pary instrukcji JUMP/LABEL lub pary instrukcji MCR/ ENDMCR. Dodatkowo, para instrukcji MCR/ ENDMCR lub inna para instrukcji JUMP/ LABEL nie mogą znajdować się obrębnie instrukcji JUMP/ LABEL bez możliwości zagnieżdżania.

Uwaga

W oprogramowaniu systemowym jednostek centralnych serii 90-30 wer. 1 dostępna jest wyłącznie nie zagnieżdżona forma instrukcji JUMP. We wszystkich nowo opracowywanych aplikacjach zalecane jest stosowanie zagnieżdżanej instrukcji JUMP.

Należy również zwrócić uwagę, że jednostki centralne serii 35x i 36x obsługują wyłącznie zagnieżdżane instrukcje skoku. Nie zagnieżdżane instrukcje skoku nie są obsługiwane przez jednostki centralne serii 35x i 36x.

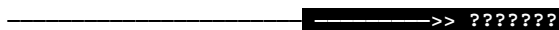
Zagnieżdżona forma instrukcji JUMP ma symbol `————N————>>LABEL01`, gdzie LABEL01 jest nazwą zagnieżdżonej instrukcji LABEL. Instrukcja ta dostępna jest w wersji 2 i wszystkich wersjach późniejszych oprogramowania Logicmaster i oprogramowania systemowego sterowników 90-30/20/Micro.

Zagnieżdżona instrukcja JUMP może być umieszczana w dowolnym miejscu w programie pod warunkiem, że jest poprawnie zagnieżdżona wewnątrz innych nie zagnieżdżonych i zagnieżdżonych instrukcji JUMP.

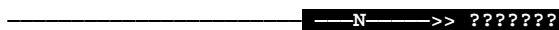
Pojedynczej, zagnieżdżanej instrukcji LABEL może odpowiadać kilka instrukcji JUMP tego samego typu. Zagnieżdżana instrukcja JUMP może powodować przejście zarówno w kierunku do przodu jak i do tyłu programu sterującego.

Obydwie formy instrukcji JUMP umieszczane są zawsze w kolumnie 9 i 10 bieżącego szczebla, po instrukcji JUMP w szczeblu nie mogą być umieszczane żadne inne elementy logiczne. Po napotkaniu takiej instrukcji, sygnał sterujący kierowany jest bezpośrednio do szczebla o zadanej etykietce.

Nie zagnieżdżona instrukcja JUMP:



Zagnieżdżona instrukcja JUMP:



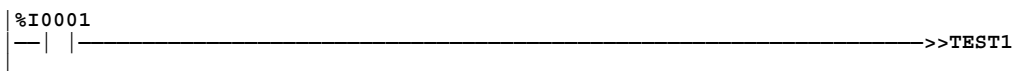
Ostrzeżenie

Aby uniknąć zapętlenia programu sterującego przy użyciu instrukcji skoku w przód i w tył, z instrukcją skoku w tył musi być związany warunek logiczny.

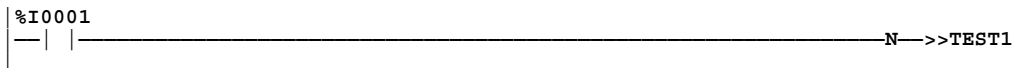
Przykłady

W zamieszczonych poniżej przykładach, po doprowadzeniu sygnału sterującego do instrukcji JUMP TEST1, jest on dalej kierowany do instrukcji LABEL TEST1.

Przykład nie zagnieżdżonej instrukcji JUMP:



Przykład zagnieżdżonej instrukcji JUMP:



LABEL

Instrukcja LABEL (etykieta) określa miejsce docelowe skoku (wywołanego przez funkcję JUMP z tą samą etykietą). Powoduje on kontynuację normalnego wykonywania programu sterującego począwszy od tej etykiety.

W programie sterującym nie mogą wystąpić dwie takie same etykiety. Również nie może zostać wykonany program, w którym występują instrukcje JUMP lub LABEL, bez odpowiedniej pary.

Oprogramowanie Logicmaster 90-30/20/Micro udostępnia dwa rodzaje funkcji etykiet: zagnieżdżane i nie zagnieżdżane. Forma nie zagnieżdżana, LABEL01, może być stosowana wyłącznie z nie zagnieżdżaną instrukcją JUMP: `—————>>LABEL01`. Forma zagnieżdżana, LABEL01, może być stosowana wyłącznie z zagnieżdżaną instrukcją JUMP: `———N——>>LABEL01`.

Instrukcja LABEL nie posiada w ogóle parametrów ani symbolu, oprócz samej etykiety, która jest ciągiem znaków alfanumerycznych.

Nie zagnieżdżona instrukcja LABEL:

```
???????:
```

Zagnieżdżona instrukcja LABEL:

```
??????? : (zagnieżdżona)
```

Przykład

W zamieszczonym poniżej przykładzie, sygnał sterujący, po dojściu do instrukcji JUMP TEST1, jest doprowadzany do szczybla LABEL TEST1.

Przykład nie zagnieżdżonej instrukcji LABEL:

```
| TEST1 :
```

Przykład zagnieżdżonej instrukcji LABEL:

```
| TEST1 : (zagnieżdżona)
```

COMMENT

Instrukcja COMMENT umożliwia wstawienie w dowolnym miejscu programu komentarza (objaśnienia dla szerebła programu sterującego). Komentarz może zawierać do 2048 znaków. W drabinie logicznej programu sterującego ma on następującą formę:

```
(* KOMENTARZ *)
```

W celu zmiany tekstu komentarza należy ustawić na nim kursor, a następnie nacisnąć klawisz F10 (Zoom). Tekst komentarza można również wydrukować.

Dłuższe komentarze można również wprowadzać z pliku utworzonego za pomocą dowolnego edytora tekstu, pracującego w systemie MS-DOS, zgodnie z przedstawionym poniżej opisem:

1. Utworzyć komentarz:
 - A. Wprowadzić tekst, aż do miejsca, w którym dołączany będzie tekst znajdujący się w pliku.
 - B. Przesunąć kursor do początku nowej linii, a następnie wprowadzić `\I` lub `\i`, oznaczenie literowe dysku, dwukropek, kartotekę lub podkartotekę zawierającą żądany plik i ostatecznie nazwę pliku, tak jak to pokazano w zamieszczonym poniżej przykładzie:

```
\I d:\text\commnt1
```

Jeżeli zarówno plik jak i kartoteka programu znajdują się na tym samym dysku, nie jest konieczne wprowadzanie oznaczenia literowego dysku.

- C. Kontynuować edycję programu lub wyjść do systemu MS-DOS.
2. Po wyjściu z pakietu do programowania, utworzyć plik tekstowy za pomocą dowolnego edytora kompatybilnego z systemem MS-DOS. Plik ten zapisać następnie pod odpowiednią nazwą i w odpowiednim miejscu, zgodnie z parametrami wprowadzonymi w instrukcji komentarza

SVCREQ

Blok funkcyjny SVCREQ służy do obsługi jednej ze specjalnych funkcji sterownika.

Tabela 12-4 Zestawienie bloków funkcyjnych SVCREQ

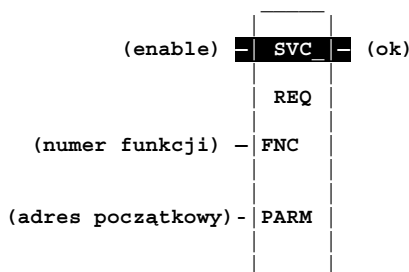
Numer	Opis
1	Zmiana/ odczyt czasu trwania cyklu pracy sterownika w trybie o stałym czasie trwania.
2	Odczyt wartości z programatora.
3	Zmiana trybu i czasu komunikacji z programatorem.
4	Zmiana trybu komunikacji systemowej oraz wartości przypisanej do generatora sygnału prostokątnego.
6	Odczyt/ zmiana liczby słów sumy kontrolnej programu sterującego.
7	Odczyt/ zmiana wskazań zegara czasu rzeczywistego.
8	Zerowanie zegara wyłączającego.
9	Odczyt czasu trwania cyklu.
10	Odczyt nazwy katalogu.
11	Odczyt identyfikatora sterownika.
12	Odczyt trybu pracy sterownika.
13	Zatrzymanie sterownika na końcu bieżącego cyklu.
14	Wymazanie komunikatów z tabeli błędów działania sterownika i układów wejść/wyjść.
15	Odczyt ostatnio zarejestrowanego błędu.
16	Odczyt wskazań zegara odmierzającego czas pracy sterownika.
18	Kontrola występowania wymuszonej zmiany wartości zmiennych wejściowych i wyjściowych.
23	Odczyt sumy kontrolnej programu sterującego i konfiguracji.
26/30	Porównanie rzeczywistej konfiguracji modułów wejść/ wyjść sterownika z zapisaną w pamięci.
29	Odczyt czasu trwania ostatniej przerwy w zasilaniu sterownika.
45	Pominięcie obsługi wejść/wyjść w następnym cyklu.
46	Szybki dostęp do statusu magistrali komunikacyjnej.

Podstawowe informacji o bloku funkcyjnym SVCREQ

Blok funkcyjny SVCREQ posiada trzy parametry wejściowe i jeden parametr wyjściowy. Po doprowadzeniu sygnału wejściowego wykonywana jest funkcja sterownika o numerze identyfikacyjnym określonym za pomocą parametru FNC. Blok parametrów odczytywany jest z pamięci sterownika, z obszaru o adresie zadanym parametrem PARM. Blok funkcyjny SVCREQ zawsze przesyła sygnał wyjściowy, chyba że podany zostanie niewłaściwy numer identyfikacyjny funkcji, błędne parametry lub adres wychodzący poza zakres pamięci sterownika. Ewentualne dodatkowe przyczyny nieprawidłowego wykonania tego bloku funkcyjnego opisano na kolejnych stronach podręcznika.

Adres zadany przez parametr PARM musi odnosić się do pamięci zorientowanej rejestrowo (%R, %AI lub %AQ). Adres ten oznacza początek bloku parametrów. Kolejne słowa 16-bitowe zawierają pozostałe parametry. Liczba parametrów zależy od typu bloku funkcyjnego (tzn. od numeru identyfikacyjnego FNC).

Obszar pamięci rozpoczynający się od adresu PARM zarezerwowany jest zarówno dla bloku parametrów wejściowych, jak i wyjściowych, przesyłanych jako wynik działania jednej z funkcji specjalnych sterownika. Parametry wejściowe i wyjściowe są więc przechowywane w tym samym miejscu.



Parametry

Parametr	Opis
enable	Sygnał wejściowy.
FNC	Stała lub adres zmiennej, określające numer identyfikacyjny jednej z funkcji specjalnych sterownika.
PARM	Adres początkowy bloku parametrów dla funkcji określonej za pomocą parametru FNC.
ok	Sygnał wyjściowy, wysyłany po poprawnym wykonaniu bloku funkcyjnego.

Dopuszczalne typy parametrów

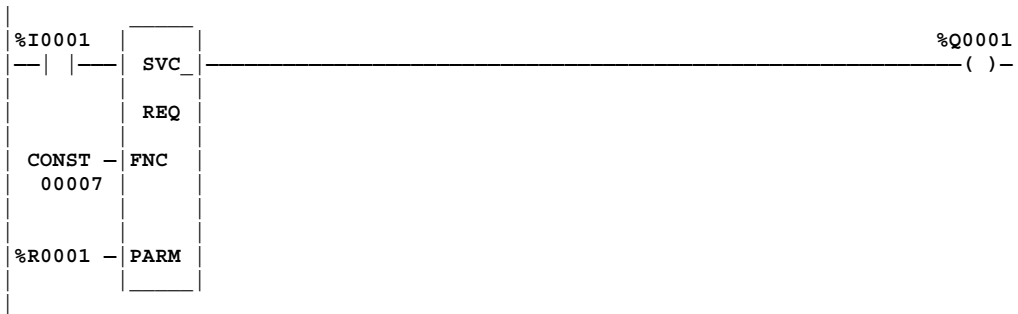
Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
FNC		•	•	•	•		•	•	•	•	•	
PARM		•	•	•	•		•	•	•	•		
ok	•											•

- Dozwolony typ parametru.

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0001 na 1 powoduje wywołanie bloku funkcyjnego SVCREQ nr 7, z blokiem parametrów rozpoczynającym się od adresu %R0001.

W przypadku pomyślnego wykonania bloku funkcyjnego, wartość zmiennej przekaźnikowej %Q0001 jest ustawiana na 1.



SVCREQ #1: Zmiana/odczyt czasu trwania cyklu pracy sterownika w trybie o stałym czasie trwania

Blok funkcyjny SVCREQ #1, dostępny w jednostkach centralnych serii 90-30 z oprogramowaniem systemowym wer. 8 i nowsze pozwala na:

- Wyłączenie trybu o stałym czasie trwania (**CONSTANT SWEEP**).
- Włączenie trybu o stałym czasie trwania (**CONSTANT SWEEP**), z pozostawieniem poprzedniej wartości czasu trwania cyklu pracy sterownika.
- Włączenie trybu o stałym czasie trwania (**CONSTANT SWEEP**) i wprowadzenie nowego czasu trwania cyklu.
- Wprowadzenie nowej wartości czasu trwania cyklu pracy sterownika.
- Sprawdzenie, czy sterownik pracuje w trybie **CONSTANT SWEEP** oraz odczyt czasu trwania cyklu pracy sterownika.

Uwaga

Spośród jednostek centralnych opisywanych w niniejszym podręczniku, blok funkcyjny SVCREQ #1 dostępny jest wyłącznie w jednostkach centralnych serii 90-30 z oprogramowaniem systemowym wer. 8.0 lub nowsza.

Blok parametrów posiada długość dwóch słów.

W celu wyłączenia trybu **CONSTANT SWEEP**, wprowadzić blok funkcyjny SVCREQ #1 z następującym blokiem parametrów:

0	adres
ignorowany	adres + 1

W celu aktywowania trybu **CONSTANT SWEEP**, wprowadzić blok funkcyjny SVCREQ #1 z następującym blokiem parametrów:

1	adres
0 lub czas trwania cyklu pracy sterownika	adres + 1

Uwaga

Jeżeli zachodzi potrzeba wprowadzenia nowej wartości czasu trwania cyklu, wartość tę należy wprowadzić w drugim słowie. Jeżeli długość cyklu programu ma pozostać bez zmian, w drugim słowie należy wprowadzić 0. Jeżeli wartość czasu trwania cyklu pracy sterownika nie została do tej pory określona, wprowadzenie wartości 0 spowoduje ustawienie wyjścia OK na 0.

W celu dokonania zmiany czasu trwania cyklu programu **bez** wybierania trybu o stałym czasie trwania, należy wprowadzić blok funkcyjny SVCREQ #1 z następującym blokiem parametrów:

2	adres
nowy czas trwania cyklu	adres + 1

W celu odczytania bieżącego czasu trwania czasu cyklu, bez wprowadzania jakichkolwiek zmian, blok funkcyjny SVCREQ #1 powinna być wywołana z następującym blokiem parametrów:

3	adres
ignorowany	adres + 1

Uwaga

Jeżeli blok funkcyjny SVCREQ #1 wywołany zostanie z blokiem parametrów, pokazanym na poprzedniej stronie, jednostki centralne z oprogramowaniem systemowym wer. 8 lub nowsza, zwrócą wartość 0 dla trybu standardowego i 1 dla trybu o stałym czasie trwania. Nie należy mylić tych wartości z wartościami parametrów wejściowych, opisanych na następnej stronie.

Blok funkcyjny zostanie pomyślnie wykonany, o ile nie wystąpi jeden z podanych poniżej przypadków:

- 1. Jako parametr określający żądane działanie do wykonania wprowadzono liczbę inną niż 0, 1, 2 lub 3.

0	Wyłączenie trybu o stałym czasie trwania (CONSTANT SWEEP).
1	Włączenie trybu o stałym czasie trwania (CONSTANT SWEEP).
2	Wprowadzenie nowej wartości czasu trwania cyklu pracy sterownika.
3	Sprawdzenie, czy sterownik pracuje w trybie CONSTANT SWEEP oraz odczyt czasu trwania cyklu sterownika. (Proszę porównać z zamieszczona powyżej Uwagą).

- 2. Wartość czasu trwania cyklu pracy sterownika jest większa od 2550 ms (2.55 sek).
- 3. Tryb o stałym czasie trwania cyklu jest włączany, jeżeli nie jest wprowadzona wartość czasu trwania cyklu pracy sterownika lub też jeśli wprowadzono dla niego wartość 0.

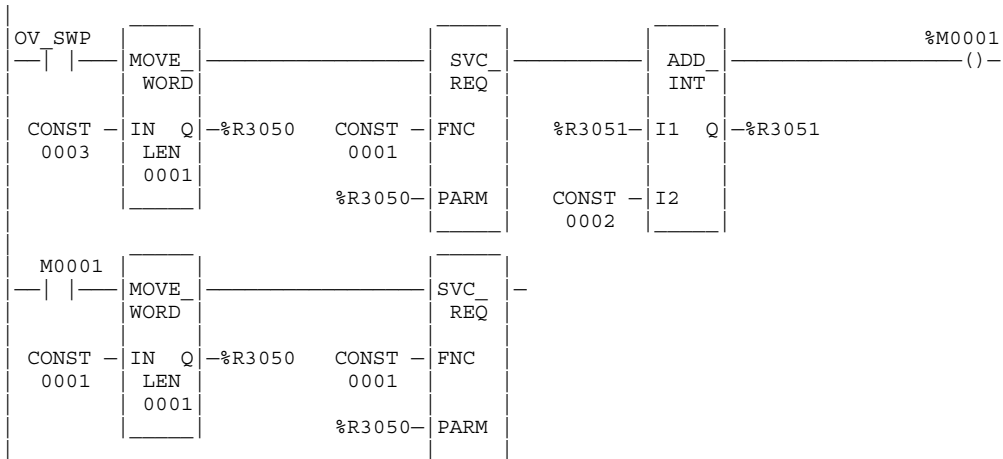
Po wykonaniu, blok funkcyjny zwraca w tym samym bloku parametrów czas trwania cyklu oraz wprowadzoną dla niego wartość.

0 = wyłączony	
1 = włączony	adres
wartość bieżąca wartość czasu trwania cyklu	adres + 1

Jeżeli słowo o adresie adres +1 ma wartość heksadecymalną FFFF, oznacza to, że nie wprowadzono żadnej wartości czasu trwania cyklu pracy sterownika.

Przykład

Zamieszczony poniżej przykład zawiera fragment programu sterującego. Jeżeli zmienna OV_SWP jest równa 1, następuje odczyt ustawionego stałego czasu trwania cyklu, następnie odczytana wartość jest zwiększana o dwie milisekundy i przesyłana z powrotem do sterownika. Blok parametrów przechowywany jest w pamięci lokalnej o adresie początkowym %R3050. Ponieważ funkcje MOVE i ADD wymagają trzech poziomych połączeń styków, w przykładzie wykorzystano wewnętrzny przekaźnik %M0001 do przechowywania wyniku wykonywania pierwszego szczebla. W każdym cyklu, w którym wartość zmiennej OV_SWP jest równa 0, zmienna %M0001 jest także równa 0.



SVCREQ #2: Odczyt wartości z programatora

Blok funkcyjny SVCREQ #2 pozwala na odczyt trybu i czasu trwania fazy komunikacji z programatorem, fazy komunikacji systemowej oraz fazy komunikacji z zadaniami wykonywanymi w tle.

Uwaga

Spośród jednostek centralnych opisywanych w niniejszym podręczniku, blok funkcyjny SVCREQ #2 dostępny jest wyłącznie w jednostkach centralnych serii 90-30 z oprogramowaniem systemowym ver. 8.0 lub nowsza.

Istnieją trzy tryby komunikacji:

Nazwa trybu	Wartość	Opis
Limited Mode (Komunikacja ograniczona czasowo)	0	Czas trwania komunikacji jest ograniczony do wartości domyślnej lub wartości zdefiniowanej za pomocą SVCREQ #3 dla komunikacji z programatorem, albo SVCREQ #4 dla komunikacji systemowej. Komunikacja jest zakończona, jeżeli zrealizowane zostały wszystkie zapytania.
Constant Mode (Stały czas trwania komunikacji)	1	Dla każdego z rodzajów komunikacji ustawiany jest tryb Realizacji wszystkich zapytań (RUN TO COMPLETION), a sterownik przełącza się pomiędzy poszczególnymi fazami komunikacji na czas równy sumie odpowiednich wartości dla każdego rodzaju komunikacji. Jeżeli dla jednego typu komunikacji wybrano tryb stałego czasu trwania (CONSTANT), tryb ten jest automatycznie wybierany dla pozostałych dwóch typów komunikacji. Jeżeli sterownik pracuje w trybie o stałym czasie trwania komunikacji (CONSTANT WINDOW) i czas trwania dla jednego z typów komunikacji nie został określony za pomocą odpowiedniego bloku funkcyjnego SVCREQ, do obliczenia czasu trwania komunikacji wykorzystywany jest czas domyślny dla danego rodzaju komunikacji.
Run to Completion (Realizacja wszystkich zadań)	2	Bez względu na fakt, czy stosowany jest domyślny czy też określony za pomocą odpowiedniego bloku funkcyjnego SVCREQ czas trwania komunikacji, komunikacja jest realizowana aż do momentu zrealizowania wszystkich zapytań.

Po ustawieniu wartości czasu na zero, komunikacja jest pomijana.

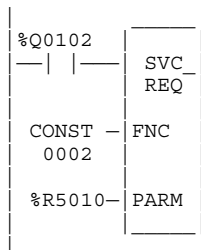
Blok parametrów posiada długość trzech słów.

	Bajt starszy	Bajt młodszy	
Komunikacja z programatorem	Tryb	Wartość w ms	adres
Komunikacja systemowa	Tryb	Wartość w ms	adres + 1
Komunikacja z zadaniami realizowanymi w tle			adres + 2

Wszystkie parametry są parametrami wyjściowymi. Blok ten nie wymaga określania jakichkolwiek parametrów w momencie jej wywołania. Parametry wyjściowe dla trzech rodzajów komunikacji podawane są w milisekundach.

Przykład

W zamieszczonym poniżej przykładzie, po każdej zmianie wartości zmiennej %Q0102 na 1, sterownik zapisuje bieżące wartości czasu trwania dla trzech rodzajów komunikacji w bloku parametrów, umieszczonym pod adresem %R5010. Dodatkowe przykłady, obrazujące zastosowanie bloku funkcyjnego SVCREQ #2 zamieszczono w opisach trzech następnych bloków funkcyjnych SVCREQ.



SVCREQ #3: Zmiana trybu i czasu trwania komunikacji z programatorem

Blok funkcyjny SVCREQ #3 pozwala na zmianę trybu komunikacji z programatorem oraz czasu trwania tej komunikacji. Zmiany zostaną uwzględnione w cyklu następującym po cyklu, w którym wywołany został ten blok funkcyjny.

Uwaga

Spośród jednostek centralnych opisywanych w niniejszym podręczniku, blok funkcyjny SVCREQ #3 dostępny jest wyłącznie w jednostkach centralnych serii 90-30 z oprogramowaniem systemowym ver. 8.0 lub nowsza.

Blok funkcyjny SVCREQ #3 wysyła sygnał wyjściowy o ile został wybrany tryb 0 (Limited), 1 (Constant) lub 2 (**RUN-TO-COMPLETION**).

Blok parametrów posiada długość jednego słowa.

W celu wyłączenia komunikacji z programatorem, wprowadzić blok funkcyjny SVCREQ #3 z następującym blokiem parametrów:

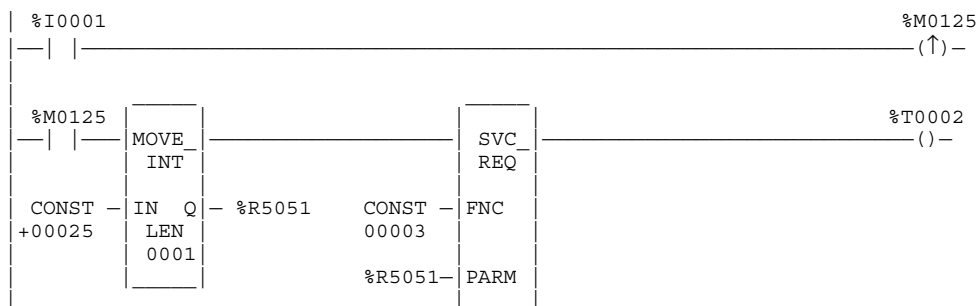
Bajt starszy	Bajt młodszy	
0	0	adres

W celu włączenia komunikacji z programatorem, wprowadzić blok funkcyjny SVCREQ #3 z następującym blokiem parametrów:

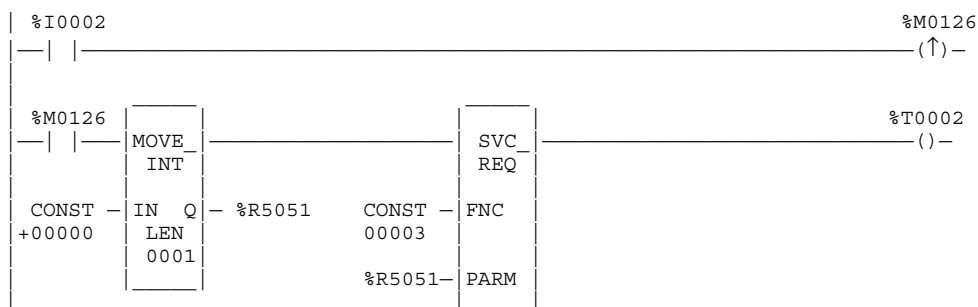
Bajt starszy	Bajt młodszy	
Tryb	Wartość z przedziału 1 do 255 ms	adres

Przykład

W zamieszczonym poniżej przykładzie, po każdej zmianie wartości zmiennej %M0125 na 1, włączana jest komunikacja z programatorem, z ustawieniem czasu na 25 ms. Blok parametrów przechowywany jest w pamięci lokalnej o adresie początkowym %R5051.



W celu wyłączenia komunikacji z programatorem, należy wywołać blok funkcyjny SVCREQ #3 z wartością zero (0). W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %M0126 na 1 powoduje włączenie komunikacji z programatorem i ustawienie wartości czasu na 0 ms. Blok parametrów przechowywany jest w pamięci lokalnej o adresie początkowym %R5051.



SVCREQ #4: Zmiana trybu i czasu trwania komunikacji systemowej

Blok funkcyjny SVCREQ #4 pozwala na zmianę trybu komunikacji systemowej oraz czasu przeznaczonych na komunikację systemową. Zmiany zostaną uwzględnione w cyklu następującym po cyklu, w którym wywołany został ten blok funkcyjny.

Uwaga

Spośród jednostek centralnych opisywanych w niniejszym podręczniku, blok funkcyjny SVCREQ #4 dostępny jest wyłącznie w jednostkach centralnych serii 90-30 z oprogramowaniem systemowym ver. 8.0 lub nowsza.

Blok funkcyjny SVCREQ #4 wysyła sygnał wyjściowy o ile został wybrany tryb 0 (Limited), 1 (Constant) lub 2 (**RUN-TO-COMPLETION**).

Blok parametrów posiada długość jednego słowa.

W celu wyłączenia komunikacji systemowej, wprowadzić blok funkcyjny SVCREQ #4 z następującym blokiem parametrów:

Bajt starszy	Bajt młodszy
0	0

 adres

W celu włączenia komunikacji systemowej, wprowadzić blok funkcyjny SVCREQ #4 z następującym blokiem parametrów:

Bajt starszy	Bajt młodszy
Tryb	Wartość z przedziału 1 do 255 ms

 adres

SVCREQ #6: Odczyt/zmiana liczby słów sumy kontrolnej programu sterującego

Blok funkcyjny SVCREQ z parametrem FNC o wartości 6 pozwala na:

- odczyt bieżącej liczby słów sumy kontrolnej programu sterującego,
- zmianę liczby słów sumy kontrolnej.

Blok funkcyjny jest poprawnie wykonywany, pod warunkiem, że jako parametr określający żądane do wykonania działanie wprowadzona zostanie liczba 0 lub 1 (proszę porównać z zamieszczonymi poniżej informacjami).

Blok parametrów ma długość 2 słów.

W celu odczytania bieżącej liczby słów sumy kontrolnej programu sterującego:

Wywołać blok funkcyjny SVCREQ #6 z następującymi parametrami:

0	adres
ignorowany	adres + 1

Po wykonaniu, blok funkcyjny zwraca w tym drugim słowie bloku parametrów wartość sumy kontrolnej. Funkcja odczytu nie ma określonego zakresu wartości, zwracana jest liczba słów, dla których obliczana jest suma kontrolna.

0	adres
liczba słów sumy kontrolnej	adres + 1

W celu ustawienia liczby słów do obliczania sumy kontrolnej:

Wywołać blok funkcyjny SVCREQ #6 z następującymi parametrami:

1	adres
liczba słów sumy kontrolnej	adres + 1

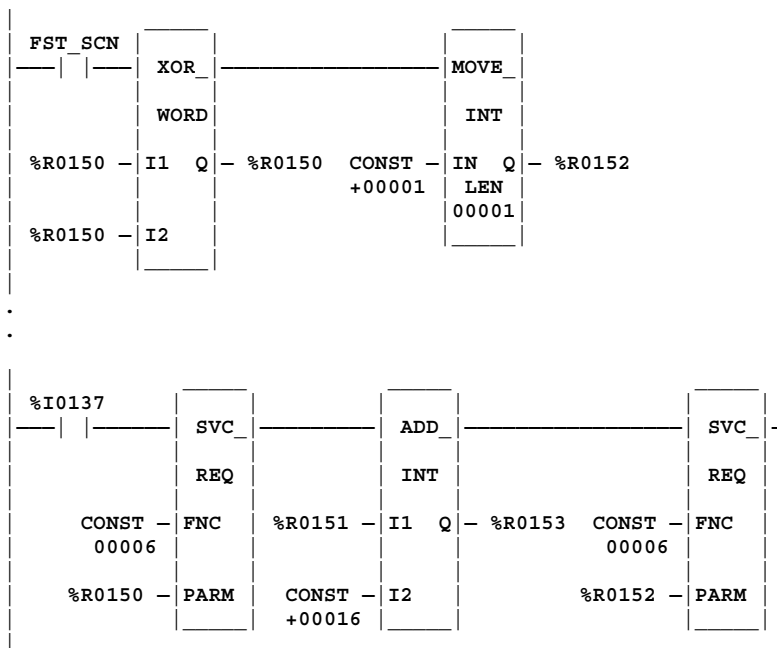
Po wprowadzeniu wartości 1, sterownik ustawia liczbę słów do obliczania sumy kontrolnej na wartość podaną jako drugie słowo bloku parametrów. Dla jednostek centralnych 331 lub 311 można wprowadzić wartość z zakresu 0 do 32.

Uwaga

Omawiany blok funkcyjny nie jest dostępny w sterownikach Micro.

Przykład

W zamieszczonym poniżej przykładzie, zmiana wartości zmiennej FST_SCN na 1 powoduje utworzenie bloku parametrów do obliczania sumy kontrolnej. W dalszej części programu, po zmianie wartości wejścia %I0137 na 1, sterownik odczytuje liczbę słów dla których obliczana jest suma kontrolna. Liczba ta jest następnie zwiększana o 16, a wynik działania funkcji ADD_INT przekazywany jest do parametru "nowa liczba słów". W następnym wywołaniu tej funkcji, nowa liczba słów do obliczania sumy kontrolnej, wprowadzana jest do sterownika.



Bloki parametrów umieszczone są pod adresem %R0150. Mają one następującą zawartość:

0 = odczyt liczby słów do obliczania sumy kontrolnej	%R0150
Bieżąca liczba słów do obliczania sumy kontrolnej	%R0151
1 = ustawianie nowej liczby słów	%R0152
nowa liczba słów do obliczania sumy kontrolnej	%R0153

SVCREQ #7: Odczyt/zmiana wskazań zegara czasu rzeczywistego

Blok funkcyjny SVCREQ #7 pozwala na odczyt/ zmianę wskazań zegara sterownika, podtrzymującego aktualną datę i czas.

Uwaga

Omawiany blok funkcyjny dostępny jest wyłącznie w jednostkach centralnych 90-30, modele 331 i wyższe, w jednostkach centralnych 28-o punktowych serii 90-30 (tzn. IC693UDR005, IC693UAA007 i IC693UDR010) oraz w jednostkach centralnych 23 punktowych sterowników serii 90 (IC693UAL006).

Blok funkcyjny jest poprawnie wykonywany, o ile nie wystąpi jeden z podanych poniżej przypadków:

1. Wartość pierwszego słowa bloku parametrów jest różna od 0 i od 1 (proszę porównać z opisem poniżej).
2. Wartość drugiego słowa bloku parametrów jest różna od 1 i od 3, co wskazuje na niemożliwy do zastosowania format zapisu.
3. Dane są zapisane w formacie innym niż wyszczególniony za pomocą drugiego słowa bloku parametrów.

Długość bloku parametrów blok funkcyjny SVCREQ #7 zależy od formatu, w którym zapisane są dane. Kod zapisu BCD (liczby dziesiętne kodowane binarnie) wymaga 6 słów, a kod ASCII wymaga 12 słów.

0 = odczyt bieżącego czasu i daty	adres
1 = ustawienie nowego czasu i daty	
1 = dane w kodzie BCD	adres + 1
3 = dane w kodzie ASCII	
DANE	adres od Parm +2 do końca bloku parametrów

Wartość 1 słowa decyduje, czy blok funkcyjny ma odczytać, czy zmienić wskazania tego zegara.

0 = czytanie
1 = zmiana

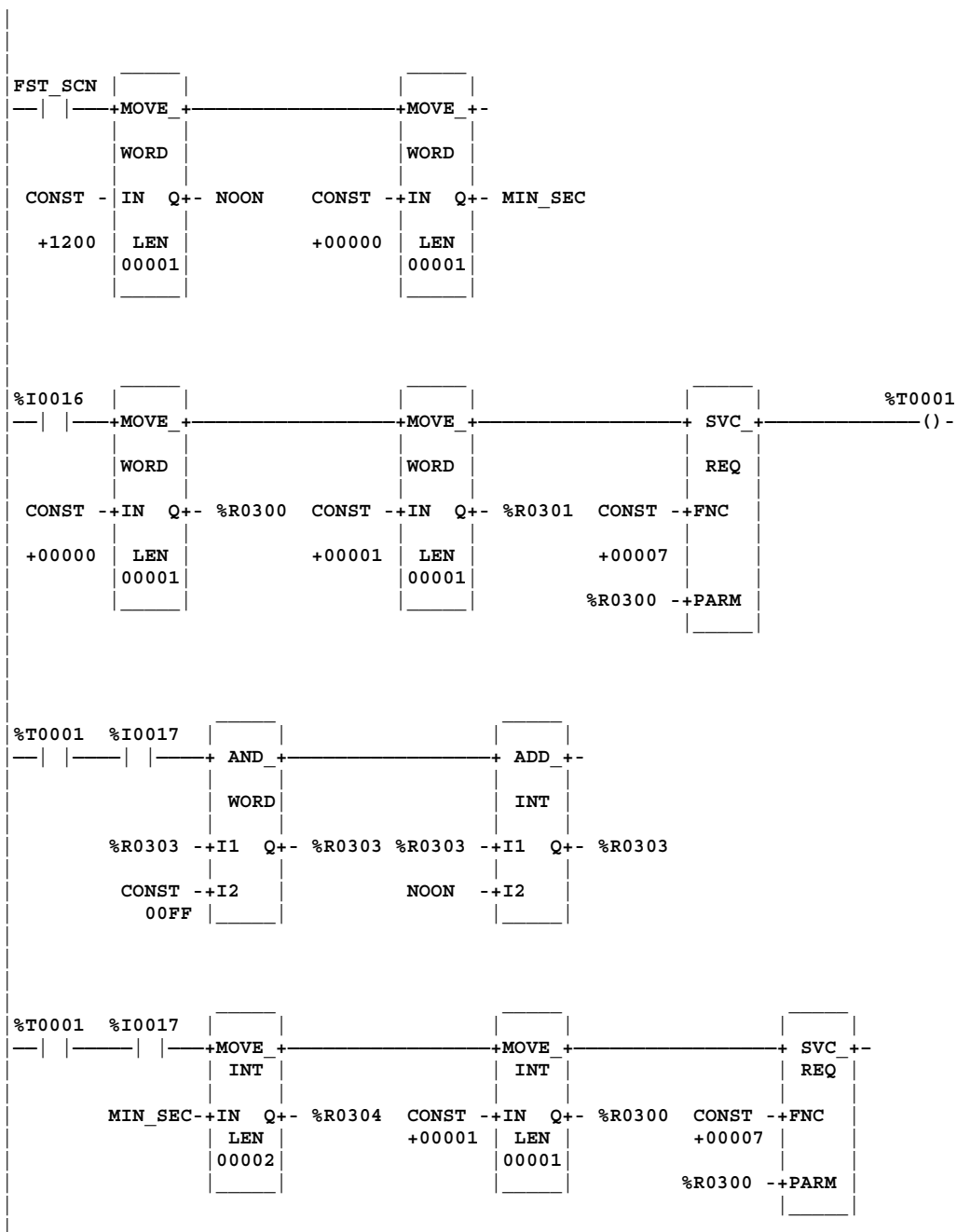
Drugie słowo określa format danych:

1 = kod BCD
3 = kod ASCII ze spacjami i znakami dwukropka (:)

Słowa od trzeciego do końca bloku danych zawierają wartości zwracane przez blok funkcyjny w przypadku odczytu (wartość pierwszego słowa bloku parametrów równa 0) lub zadane przez użytkownika w przypadku ustawiania nowego czasu i daty (wartość pierwszego słowa bloku parametrów równa 1). W obydwu przypadkach, format słów z danymi jest taki sam. W czasie czytania daty i czasu, słowa (adres+2) do (adres + 8) bloku parametrów są ignorowane.

Przykład

W zamieszczonym poniżej przykładzie tworzony jest blok parametrów, który ma za zadanie odczytanie bieżącej daty i czasu, a następnie ustawienie wskaźnika zegara na godzinę 12:00:00 w formacie BCD. Blok parametrów umieszczony jest w obszarze danych globalnych %R0300. Tablica NOON została wcześniej zadeklarowana w programie, zawiera ona wartości 12, 0 i 0. (Tablica NOON musi zawierać również dane zapisane pod adresem %R0300.) Format zapisu BCD wymaga zarezerwowania sześciu sąsiadujących komórek pamięci na blok parametrów.



Zawartość bloku danych

Blok danych zawiera dokładną datę i czas, zapisane wg podanych poniżej reguł. Dla obydwu formatów zapisu danych:

- Godziny są zapisywane w konwencji 24-godzinnej;
- Dni tygodnia zapisywane są w konwencji numerycznej:

Wartość	Dzień tygodnia
1	Poniedziałek
2	Wtorek
3	Środa
4	Czwartek
5	Piątek
6	Sobota
7	Niedziela

Zmiana/ odczyt czasu w formacie BCD:

W formacie wykorzystującym kod zapisu BCD poszczególne elementy daty i czasu zajmują po jednym bajcie. Format taki wymaga zastosowania sześciu słów. Ostatni bajt szóstego słowa nie jest używany. Podczas ustawiania daty i czasu bajt ten jest ignorowany, natomiast przy odczytywaniu bieżącej daty i czasu bajt ten jest wypełniany zerami (tzn. ma wartość 00 w kodzie BCD).

Bajt starszy	Bajt młodszy	
1 = zmiana	lub 0 = odczyt	adres
1		adres + 1
miesiąc	rok	adres + 2
godziny	dzień miesiąca	adres + 3
sekundy	minuty	adres + 4
(puste pola)	dzień tygodnia	adres + 5

Przykładowy, wyjściowy blok parametrów:
Odczyt daty i czasu w formacie BCD
(Niedziela, 3 lipca 1988, godz. 14:45:30)

0	
1	
07	88
14	03
30	45
00	01

W celu zmiany/odczytu daty i czasu w formacie ASCII ze spacjami i znakami dwukropka (:)

W formacie tym każda cyfra daty i czasu zapisywane są w kodzie ASCII. Dodatkowo, do bloku danych włączone są spacje i dwukropki, aby umożliwić przesłanie bloku danych w niezmienionej formie na ekran lub drukarkę. Format taki wymaga zastosowania 12 słów.

Bajt starszy		Bajt młodszy		
1 = zmiana	lub	0 = odczyt		adres
3				adres + 1
rok		rok		adres +2
miesiąc		(spacja)		adres +3
(spacja)		miesiąc		adres +4
dzień miesiąca		dzień miesiąca		adres +5
godziny		(spacja)		adres +6
:		godziny		adres +7
minuty		minuty		adres +8
sekundy		:		adres +9
(spacja)		sekundy		adres +10
dzień tygodnia		dzień tygodnia		adres +11

Przykładowy, wyjściowy blok parametrów:
 Odczyt daty i czasu w formacie ASCII
 (poniedziałek, 2 październik 1989, 23:13:00)

0	
3	
39	38
31	20
20	30
32	30
32	20
3A	33
33	31
30	3A
20	30
32	30

SVCREQ #8: Zerowanie zegara wyłączającego

Blok funkcyjny SVCREQ #8 pozwala na wyzerowanie zegara wyłączającego w czasie trwania cyklu.

Uwaga

Spośród jednostek centralnych opisywanych w niniejszym podręczniku, blok funkcyjny SVCREQ #8 dostępny jest wyłącznie w jednostkach centralnych serii 90-30 z oprogramowaniem systemowym ver. 8.0 lub nowsza.

Po upływie nastawionego na tym zegarze czasu, sterownik jest resetowany. Blok funkcyjny pozwala na zmianę wskazań tego zegara w czasie czasochłonnych zadań (na przykład w czasie oczekiwania na odpowiedź z systemu komunikacji).

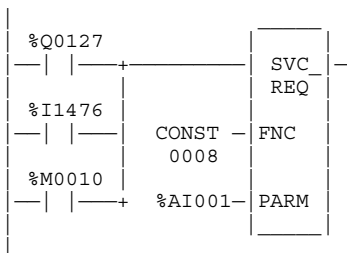
Ostrzeżenie

Należy upewnić się, że wyzerowanie pracy zegara wyłączającego nie będzie miało ujemnego wpływu na sterowany proces.

Blok funkcyjny nie posiada bloku parametrów, pomimo tego, pakiet programistyczny wymaga wprowadzenia wartości parametru PARM. Należy wprowadzić dowolną wartość, nie będzie ona wykorzystywana.

Przykład

W zamieszczonym poniżej przykładzie, zmiana wartości zmiennej %Q0127 lub wewnętrznego przełącznika %M0010 na 1 powoduje wyzerowanie zegara wyłączającego.



SVCREQ #9: Odczyt czasu trwania cyklu

Blok funkcyjny SVCREQ #9 pozwala na odczyt czasu w milisekundach, który upłynął od momentu rozpoczęcia cyklu. Dane są zapisywane w formacie 16 bitowych słów.

Uwaga

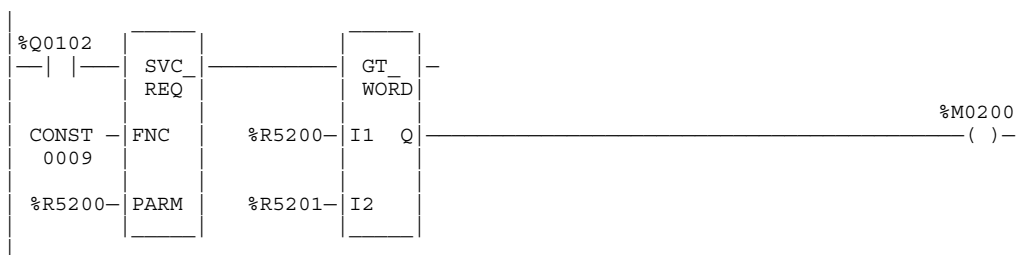
Spośród jednostek centralnych opisywanych w niniejszym podręczniku, blok funkcyjny SVCREQ #9 dostępny jest wyłącznie w jednostkach centralnych serii 90-30 z oprogramowaniem systemowym ver. 8.0 lub nowsza.

Blok parametrów zawiera wyłącznie parametry wyjściowe, ma on długość jednego słowa.

czas od rozpoczęcia cyklu adres

Przykład

W zamieszczonym poniżej przykładzie, do zmiennej %R5200 zapisywany jest czas, który upłynął od momentu rozpoczęcia cyklu. Jeżeli jego wartość jest większa od wartości zmiennej %R5201, następuje ustawienie wewnętrznego przekaźnika %M0200 na 1.



SVCREQ #11: Odczyt identyfikatora sterownika

Blok funkcyjny SVCREQ #11 pozwala na odczytanie identyfikatora sterownika serii 90, wykonującego program.

Uwaga

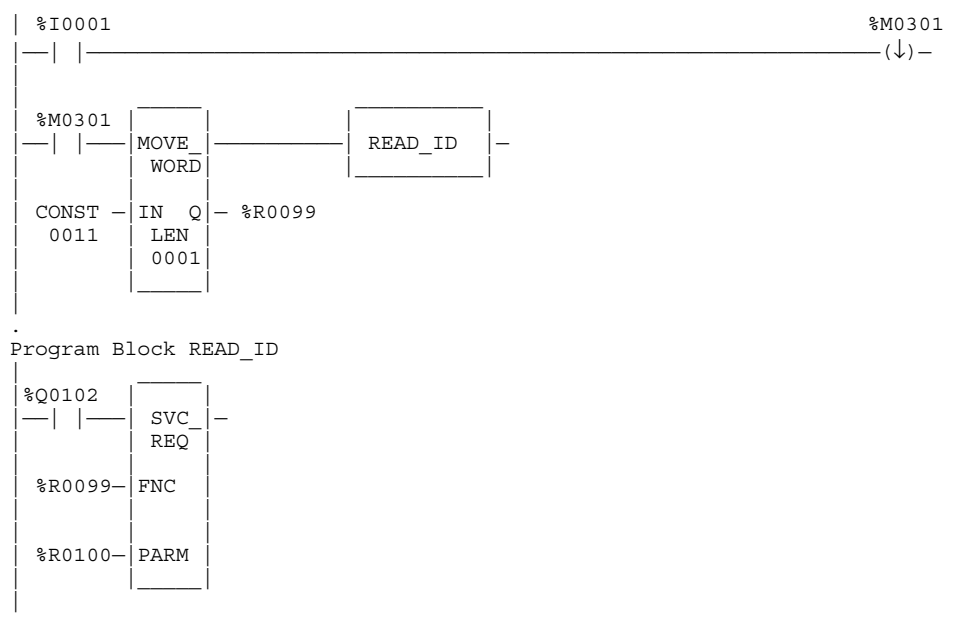
Spośród jednostek centralnych opisywanych w niniejszym podręczniku, blok funkcyjny SVCREQ #11 dostępny jest wyłącznie w jednostkach centralnych serii 90-30 z oprogramowaniem systemowym ver. 8.0 lub nowsza.

Blok parametrów wyjściowych posiada długość czterech słów. Zwraca on osiem znaków ASCII, ostatni z nich jest pusty (00h). Jeżeli długość identyfikatora sterownika jest krótsza od siedmiu znaków, na końcu dołączane są znaki puste.

Bajt młodszy		Bajt starszy		
znak 1		znak 2		adres
znak 3		znak 4		adres + 1
znak 5		znak 6		adres +2
znak 7		00		adres +3

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0001 na 0, powoduje zapisanie wartości 11 do rejestru %R0099. Wartość ta jest numerem identyfikacyjnym bloku funkcyjnego SVCREQ #11 do odczytu nazwy sterownika. Następnie wywoływany jest blok programu READ_ID, którego zadaniem jest odczytanie rzeczywistej nazwy sterownika. Blok parametrów umieszczony jest pod adresem %R0100.



SVCREQ #12: Odczyt trybu pracy sterownika

Blok funkcyjny SVCREQ #12 pozwala na odczytanie bieżącego statusu trybu pracy (**RUN**) jednostki centralnej sterownika.

Uwaga

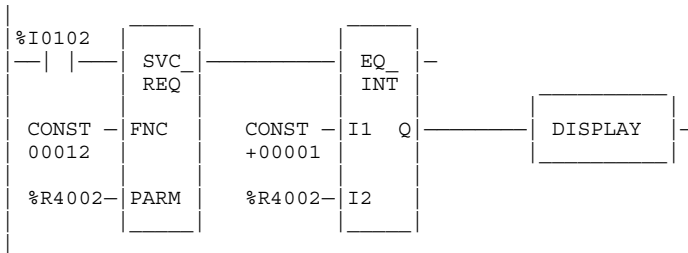
Spośród jednostek centralnych opisywanych w niniejszym podręczniku, blok funkcyjny SVCREQ #12 dostępny jest wyłącznie w jednostkach centralnych serii 90-30 z oprogramowaniem systemowym ver. 8.0 lub nowsza.

W bloku parametrów zapisywane są wyłącznie parametry wyjściowe, ma on długość jednego słowa.

1 = praca/ wyjścia nieaktywne	adres
2 = praca/ wyjścia aktywne	

Przykład

W zamieszczonym poniżej przykładzie odczytywany jest i zapisywany do zmiennej %R4002 status trybu pracy sterownika. Jeżeli stwierdzony zostanie status praca/ włączony następuje wywołanie bloku programu DISPLAY.



SVCREQ #14: Kasowanie komunikatów z tablicy błędów działania sterownika i układów wejść/wyjść

Blok funkcyjny SVCREQ z parametrem FNC o wartości 14 służy do kasowania wszystkich informacji z tablicy błędów działania sterownika lub z tabeli błędów działania układów wejść/wyjść. Blok ten wysyła sygnał wejściowy pod warunkiem, że jako parametr określający żądane do wykonania działanie wprowadzona zostanie liczba 0 lub 1 (proszę porównać z zamieszczonymi poniżej informacjami).

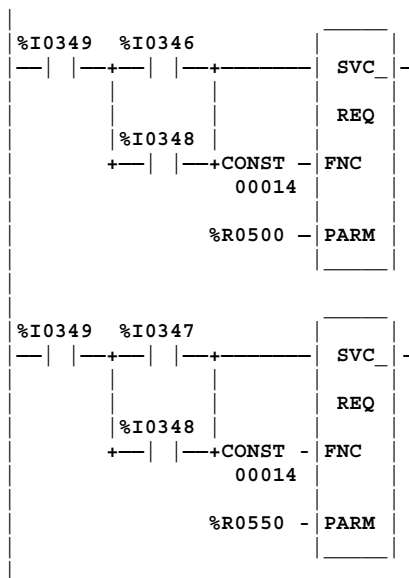
Długość bloku parametrów wynosi 1 słowo. Zawiera on wyłącznie parametr wejściowy.

0 = kasowanie zawartości tabeli błędów działania sterownika	adres
1 = kasowanie zawartości tabeli błędów działania układów wejść/ wyjść	

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennych %I0346 i %I0349 powoduje wykasowanie komunikatów z tabeli błędów działania sterownika. Jeżeli zmiennne wejściowe %I0347 i %I0349 mają wartość równą 1, następuje wykasowanie komunikatów z tabeli błędów działania układów wejść/wyjść. Jeżeli zmiennne wejściowe %I0348 i %I0349 mają wartość równą 1, kasowane są komunikaty w obydwu tabelach błędów.

Blok parametrów dla tabeli błędów działania sterownika umieszczony jest pod adresem %R0500, a blok parametrów dla tabeli błędów działania układów wejść/wyjść umieszczony jest pod adresem %R0550. Wartości wpisywane są do obydwu tych bloków parametrów w innym miejscu programu.



SVCREQ #15: Odczyt ostatnio zarejestrowanego komunikatu o błędzie działania

Blok funkcyjny SVCREQ z parametrem FNC o wartości 15 służy do odczytu ostatnio zarejestrowanego komunikatu o błędzie sterownika lub układów wejść/wyjść. Blok funkcyjny zawsze wysyła sygnał wejściowy pod warunkiem, że jako parametr określający żądane do wykonania działanie wprowadzona zostanie liczba 0 lub 1 (proszę porównać z zamieszczonymi poniżej informacjami). (Dodatkowe informacje o tabelach błędów podano w Rozdziale 3 "Błędy").

Blok parametrów zajmuje 22 słowa. Format parametrów wejściowych jest następujący:

0 = Odczyt ostatnio zarejestrowanego błędu z tabeli błędów działania sterownika	adres
1 = Odczyt ostatnio zarejestrowanego błędu z tabeli błędów działania układów wejścia/ wyjścia	

Format parametrów wyjściowych zapisywanych pod adresami od "PARM + 1" do "PARM + 21" zależy od tabeli, z której czytany jest komunikat.

Format bloku parametrów wyjściowych dla tabeli błędów działania sterownika

Bajt młodszy	Bajt starszy	
0		
Długość bloku dodatkowych informacji o błędzie		adres + 1
Pole wolne		adres +2
Adres wystąpienia błędu w działaniu sterownika		adres +3
		adres +4
Grupa błędu i kategoria błędu		adres +5
Kod błędu		adres +6
		adres +7
		adres +8
		adres +9
		adres +10
		adres +11
Blok dodatkowych informacji o błędzie		adres +12
		adres +13
		adres +14
		adres +15
		adres +16
		adres +17
		adres +18
		adres +19
Czas i data wystąpienia błędu		adres +20
		adres +21

Format bloku parametrów wyjściowych dla tabeli błędów działania układów wejść/wyjść

Bajt młodszy	Bajt starszy	
1		
Długość bloku dodatkowych informacji o błędzie		adres + 1
Adres wystąpienia błędu		adres +2
		adres +3
		adres +4
Adres wystąpienia błędu w działaniu układów wejść/wyjść		adres +5
		adres +6
Przynależność błędu do określonej grupy błędów oraz waga błędu i związana z nią reakcja systemu na jego wystąpienie		adres +7
Kategoria błędu	Typ błędu	adres +8
Opis błędu		adres +9
		adres +10
		adres +11
Blok szczegółowych informacji o błędzie		adres +12
		adres +13
		adres +14
		adres +15
		adres +16
		adres +17
		adres +18
		adres +19
Czas i data wystąpienia błędu		adres +20
		adres +21

Przykład 2

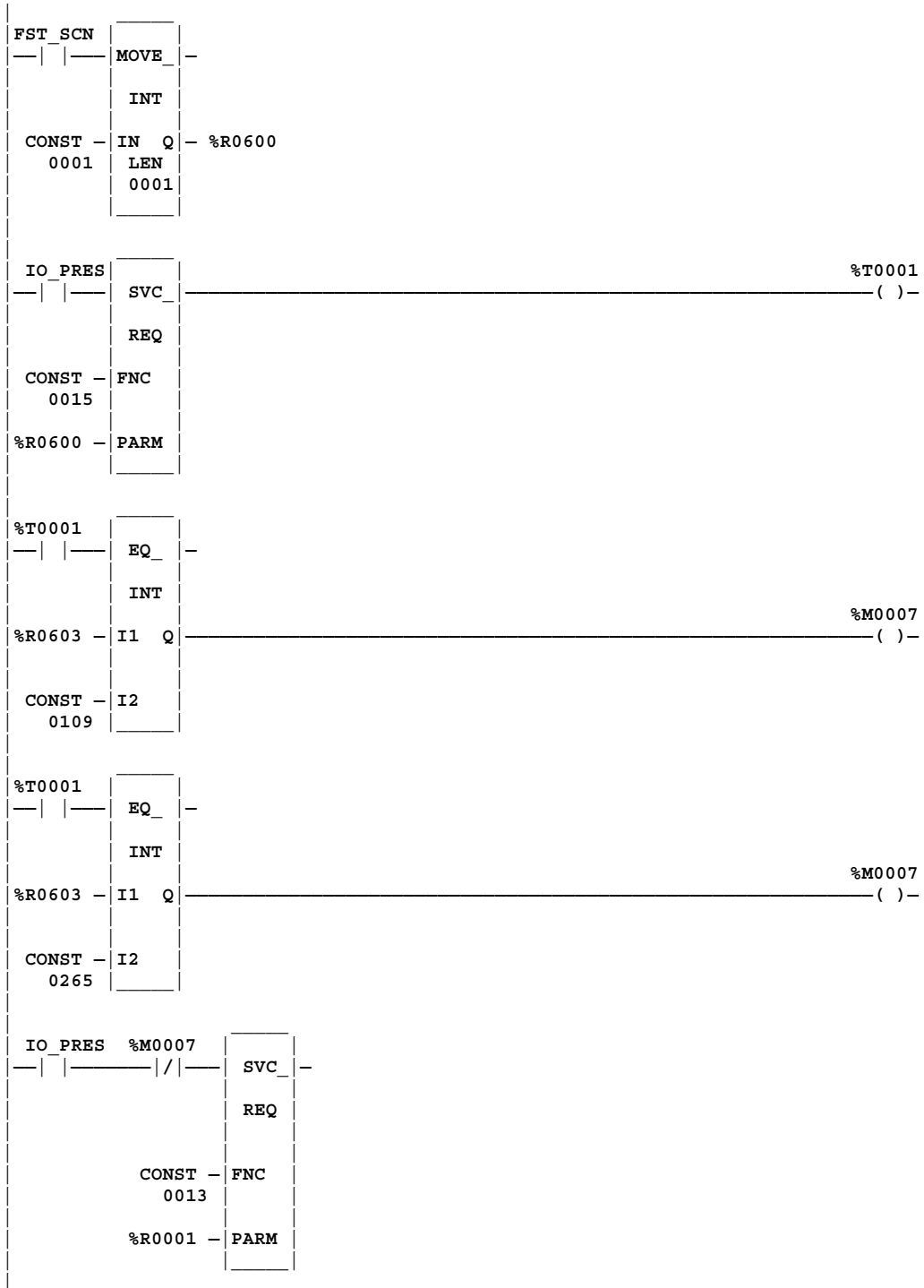
W następnym przykładzie, w przypadku wystąpienia jakiegokolwiek błędu działania modułu wejść i wyjść, za wyjątkiem wystąpienia błędów w modułach umieszczonych w gniazdach 1 i 9 kasety 0, sterownik jest zatrzymywany. Jeżeli wystąpi błąd w jednym z tych dwóch modułów, praca systemu jest kontynuowana. W pierwszym cyklu ustawiany jest parametr określający typ tabeli błędów działania. Jeżeli zmienna przypisana do styku IO_PRES ma wartość 1, informuje to o wprowadzeniu nowej pozycji do tabeli błędów działania układów wejść/wyjść. Sterownik ustawia na 1 zmienną przypisaną do styku otwartego w pierwszym cyklu, realizowanym po wprowadzeniu pozycji do tabeli błędów. Jeżeli błędy są rejestrowane w dwóch kolejnych cyklach, wartość zmiennej przypisanej do styku otwartego jest ustawiana w dwóch dalszych cyklach.

W przykładzie tym wykorzystywany jest blok parametrów umieszczony pod adresem %R0600. Po wykonaniu bloku funkcyjnego SVCREQ, piąte, szóste i siódme słowo bloku parametrów zawierają adres modułu wejść i wyjść, w którym wystąpił błąd.

1		%R0600
Długość bloku dodatkowych informacji o błędzie		%R0601
Adres wystąpienia błędu		%R0602
Numer kasety	Numer gniazda	%R0603
Nr szyny wejść/wyjść	Adres szyny	%R0604
Adres punktu		%R0605

Blok dodatkowych informacji o błędzie

W programie tym, bloki EQ_INT porównują adresy kasety/ gniazda zapisane w tabeli z stałymi heksadecymalnymi. Wartość wewnętrznego przełącznika %M0007 jest ustawiana na 1, jeżeli kasetka/ gniazdo dla których sygnalizowany jest błąd spełniają kryteria podane powyżej. Jeżeli wartość zmiennej przełącznikowej %M0007 jest równa 1, styki otwarte są rozwierane, co zapobiega zatrzymaniu sterownika. Z drugiej strony, jeżeli wartość zmiennej %M0007 jest równa 0, informuje to o wystąpieniu błędu w innym module, styki otwarte są zwierane i następuje zatrzymanie sterownika.



SVCREQ #16: Odczyt wskazań zegara odmierzającego czas pracy sterownika

Blok funkcyjny SVCREQ z parametrem FNC o wartości 16 pozwala na odczyt bieżącego stanu zegara odmierzającego czas pracy sterownika. Zegar ten odmierza w sekundach czas, który upłynął od momentu ostatniego włączenia zasilania sterownika. Zakres wskazań tego zegara wynosi ok.100 lat.

Blok parametrów zawiera wyłącznie parametry wyjściowe. Posiada on długość trzech słów.

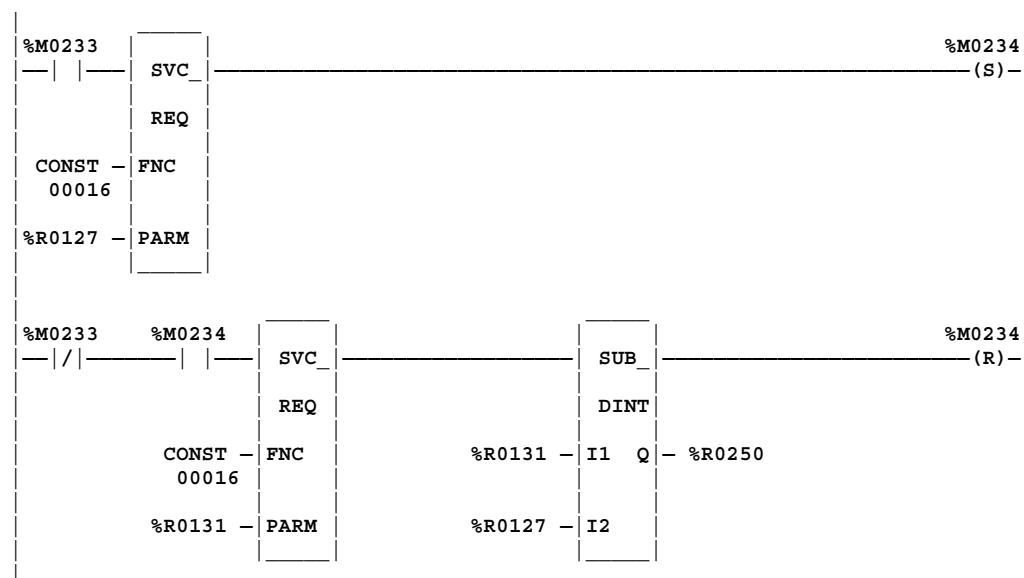
Liczba sekund, które upłynęły od momentu włączenia zasilania sterownika (niższy rząd wielkości)	adres
Liczba sekund, które upłynęły od momentu włączenia zasilania sterownika (wyższy rząd wielkości)	adres + 1
Liczba setnych części odmierzanej aktualnie sekundy	adres + 2

Pierwsze dwa słowa podają czas w sekundach. Ostatnie słowo zawiera liczbę setnych części odmierzanej aktualnie sekundy.

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej przekaźnikowej %M0233 z 0 na 1 powoduje odczytanie i zapisanie do zmiennej %M0234 wskazań tego zegara. Zmiana na 0 powoduje powtórne odczytanie tej wartości. Różnica pomiędzy tymi wartościami jest następnie zapisywana do pamięci pod adresem %R0250.

W pierwszym odczycie dane są odczytywane z obszaru określonego adresem %R0127, a drugim z obszaru określonego adresem %R0131. W czasie obliczania ignorowana jest liczba setnych części sekundy oraz fakt, że typ DINT jest w rzeczywistości liczbą całkowitą ze znakiem. Uzyskiwany jest prawidłowy wynik obliczeń, o ile czas od momentu włączenia nie przekroczył 50 lat.



SVCREQ #18: Kontrola występowania wymuszeń zmiany wartości zmiennych wejściowych i wyjściowych

Blok funkcyjny SVCREQ z parametrem FNC o wartości 18 służy do sprawdzenia, czy występują zmienne, których wartość została wymuszona i zablokowana (OVERRIDE).

Uwaga

Omawiany blok funkcyjny dostępny jest **wyłącznie w** jednostkach centralnych 331 i wyższych.

Długość bloku parametrów wynosi 1 słowo. Zawiera on wyłącznie parametr wejściowy.

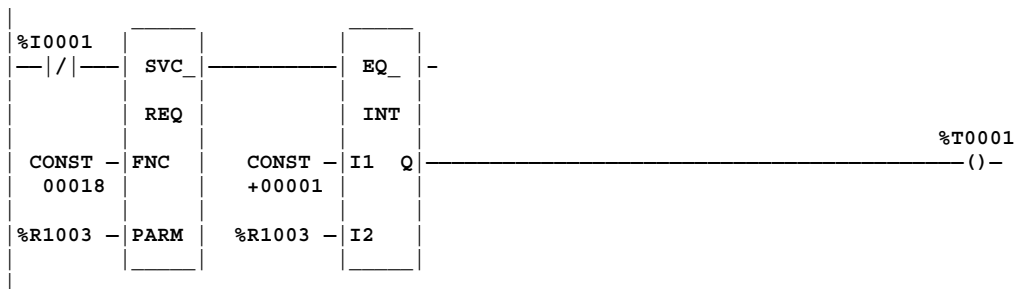
0 =	brak zmiennych, których wartość została wymuszona i zablokowana	adres
1 =	występowanie zmiennych, których wartość została wymuszona i zablokowana	

Uwaga

Blok funkcyjny SVCREQ #18 podaje informacje wyłącznie dla zmiennych typu %I i %Q.

Przykład

W zamieszczonym poniżej przykładzie odczytywany jest i zapisywany do zmiennej %R1003 status wymuszenia i zablokowania wartości zmiennych. Jeżeli występują zmienne, których wartość jest wymuszona i zablokowana, wartość zmiennej wyjściowej %T0001 ustawiana jest na 1.



SVCREQ #23: Odczyt sumy kontrolnej programu sterującego i konfiguracji

Blok funkcyjny SVCREQ z parametrem FNC o wartości 23 służy do odczytu sumy kontrolnej programu sterującego i konfiguracji. Wyjście bloku funkcyjnego jest zawsze ustawiane na 1 po uaktywnieniu bloku funkcyjnego, a blok informacji wyjściowych rozpoczyna się pod adresem podanym jako parametr PARM bloku.

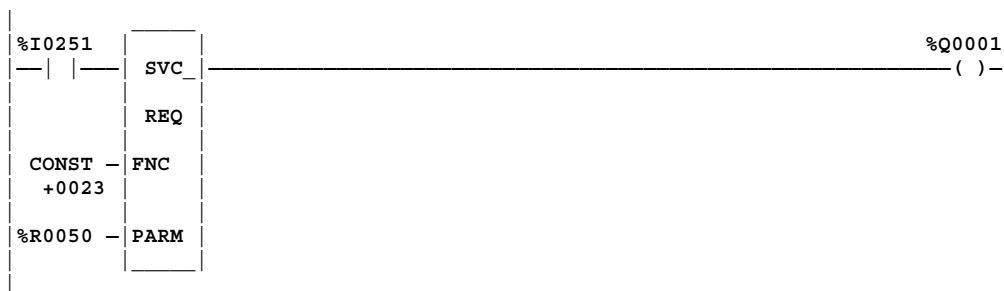
W czasie zapisywania programu (konfiguracji) w trybie **RUN**, sumy kontrolne mogą nie być uaktualnione do czasu zakończenia operacji. W celu określenia, czy sumy kontrolne są poprawne przewidziano dwa wskaźniki, umieszczone na samym początku parametrów wyjściowych bloku.

Blok parametrów wyjściowych o długości 12 słów posiada następujący format:

Suma kontrolna programu sterującego (0 = nie poprawna, 1 = poprawna)	adres
Suma kontrolna konfiguracji (0 = nie poprawna, 1 = poprawna)	adres + 1
Liczba bloków programu (włącznie z blokiem głównym MAIN)	adres +2
Rozmiar programu sterującego w bajtach (2 słowa)	adres +3
Suma kontrolna programu sterującego	adres +5
32-bitowa suma kontrolna typu CRC programu sterującego	adres +6
Rozmiar konfiguracji w bajtach	adres +8
Suma kontrolna konfiguracji	adres +9
32-bitowa suma kontrolna typu CRC dla konfiguracji (2 słowa)	adres +10

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej wejściowej %I0251 na 1 powoduje zapisanie do bloku parametrów sumy kontrolnej programu sterującego oraz ustawienie zmiennej przekaźnika %Q0001 na 1. Blok parametrów umieszczony jest pod adresem %R0050.



SVCREQ #26/30: Porównanie rzeczywistej konfiguracji modułów wejść/ wyjść sterownika z zapisaną w pamięci

Blok funkcyjny SVCREQ z parametrem FNC o wartości 26 lub 30 (obydwa parametry realizują to samo zadanie) służy do sprawdzenia, jakie moduły są aktualnie zainstalowane w sterowniku i do porównania rzeczywistej konfiguracji z zapisaną w pamięci, generując alarmy związane z zainstalowaniem i usunięciem modułu oraz niezgodnością konfiguracji. Blok funkcyjny zapisuje błędy w tabeli błędów sterownika i w tabeli błędów działania układów wejść/wyjść.

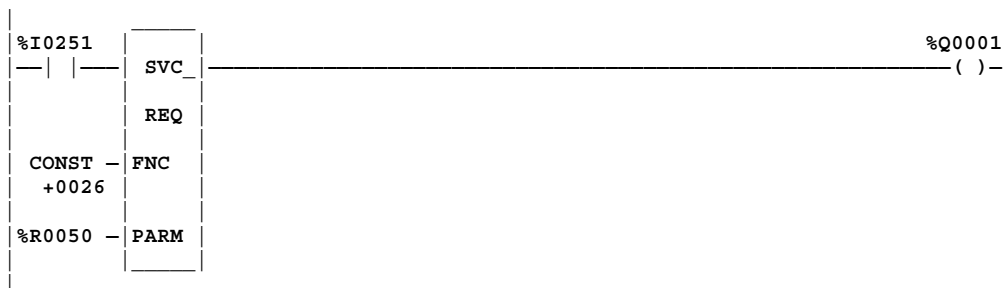
Blok SVCREQ #26/30 nie posiada parametrów i zawsze przesyła sygnał wyjściowy.

Uwaga

Czas wykonania tego bloku funkcyjnego SVCREQ zależy od liczby zarejestrowanych błędów. Z tego powodu, czas wykonania bloku funkcyjnego SVCREQ ulega wydłużeniu w przypadku, gdy błędy zarejestrowane zostały w większej liczbie modułów.

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej %I0251 na 1 powoduje sprawdzenie faktycznie zainstalowanych modułów i porównanie ich z wprowadzoną konfiguracją. Po zakończeniu wykonywania bloku funkcyjnego SVCREQ, wartość zmiennej wyjściowej %Q0001 jest ustawiana na 1.



Uwaga

Omawiany blok funkcyjny nie jest dostępny w sterownikach Micro.

SVCREQ #29: Zegar czasu trwania przerwy w zasilaniu sterownika

Blok funkcyjny SVCREQ z parametrem FNC o wartości 29 umożliwia odczyt czasu trwania ostatniej przerwy w zasilaniu sterownika. Sygnał wyjściowy jest zawsze wysyłany, a blok informacji wyjściowych rozpoczyna się pod adresem podanym jako parametr PARM bloku.

Uwaga

Omawiany blok funkcyjny dostępny jest wyłącznie w jednostkach centralnych 331 i wyższych.

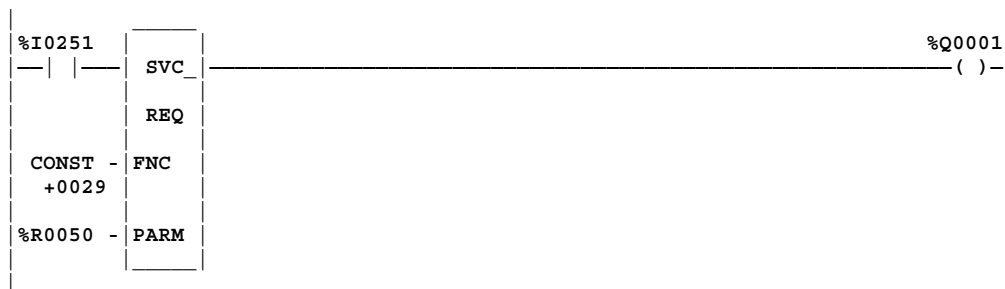
Blok parametrów zawiera jedynie parametry wyjściowe. Posiada on długość trzech słów.

Czas trwania przerwy w zasilaniu w sekundach (młodszy bajt)	adres
Czas trwania przerwy w zasilaniu w sekundach (starszy bajt)	adres + 1
Impulsy czasu o długość 100 mikrosekund	adres +2

Pierwsze dwa słowa podają czas w sekundach, który upłynął od momentu wyłączenia zasilania. Ostatnie słowo jest to pozostały czas, podawany w jednostkach o długości 100 mikrosekund. Gdy sterownik nie jest w stanie prawidłowo oszacować czasu, czas zostaje ustawiony na zero. Zdarza się to przy włączeniu zasilania sterownika przy wciśniętym klawiszu CLR M/T na ręcznym programatorze. Może to również wystąpić po zadziałaniu zegara wyłączającego.

Przykład

W zamieszczonym poniżej przykładzie, każda zmiana wartości zmiennej wejściowej %I0251 na 1 powoduje zapisanie do bloku parametrów czasu, który upłynął od momentu wyłączenia zasilania oraz ustawienie zmiennej przełącznika %Q0001 na 1. Blok parametrów umieszczony jest pod adresem %R0050.



SVCREQ #45: Pominięcie obsługi wejść/wyjść w następnym cyklu

Blok funkcyjny SVC_REQ #45 pozwala na pominięcie obsługi wejść i wyjść w następnym cyklu pracy sterownika. Jakakolwiek zmiana zmiennych przypisanych do wyjść w cyklu, w którym wykonywany jest blok SVC_REQ #45 nie zostaje odzwierciedlona na wyjściach fizycznego modułu. Jakakolwiek zmiana danych o stanie wejść fizycznych modułu nie jest odzwierciedlana w odpowiednich zmiennych przypisanych do wejść, w cyklu następującym po cyklu, w którym wywołano blok funkcyjny SVC_REQ #45.

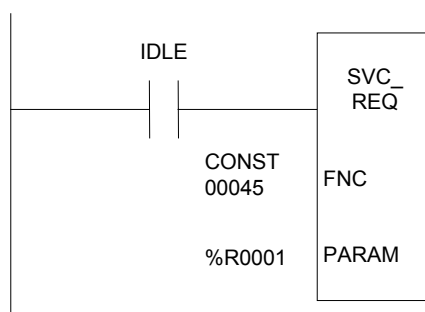
Blok ten nie posiada żadnych parametrów.

Uwaga

Blok funkcyjny SVC_REQ #45 nie ma żadnego wpływu na sposób działania bloku funkcyjnego DO_IO. Blok funkcyjny DO_IO aktualizuje Wejścia/Wyjścia, nawet jeżeli zostanie wywołany w tym samym programie co blok SVC_REQ #45.

Przykład

W zamieszczonym poniżej przykładzie, doprowadzenie sygnału do styku "Idle" powoduje pominięcie obsługi wejść/wyjść w następnym cyklu.



SVCREQ #46: Szybki dostęp do statusu

Blok funkcyjny SVCREQ #46 pozwala na wywołanie jednej z następujących funkcji:

1. Odczytanie słowa dodatkowych danych o statusie jednego z inteligentnych modułów.
2. Zapisanie słowa dodatkowych danych o statusie jednego z inteligentnych modułów.
3. Odczyt/ zapis: Odczyt słowa dodatkowych danych o statusie jednego z inteligentnych modułów i zapisanie wartości z zakresu 0-15 do tego samego modułu, w czasie jednej operacji.

Uwagi:

Blok ten dostępny jest wyłącznie w niektórych modułach. Obecnie, blok ten obsługiwany jest wyłącznie przez moduł DSM (Digital Servo Module), wersja 312, który nie jest oferowany w momencie przygotowywania niniejszego podręcznika. Planowane jest jednak, że zostanie wkrótce wprowadzony do oferty.

Bloki funkcyjne COMMREQ i DO_IO nie powinny być wywoływane dla danego modułu (modułów) w cyklu, w którym wywoływana jest blok funkcyjny do zapisu danych, ponieważ może to prowadzić do utraty zapisywanych danych.

Dwa bloki funkcyjne zapisujące do modułu (zapis lub zapis/ odczyt) nie powinny być wywoływane w tym samym cyklu, ponieważ może to prowadzić do utraty danych, które są zapisywane jako pierwsze.

Blok parametrów posiada zmienną długość. Jest on opisany poniżej. Pierwsze słowo bloku parametrów określa funkcję, która ma zostać wykonana i ma następujący format:

1 = Odczyt dodatkowych danych	adres
2 = Zapis dodatkowych danych	
3 = Odczyt/ zapis dodatkowych danych	

Odczyt dodatkowych danych o statusie (Funkcja #1)

Funkcja ta odczytuje dodatkowe dane o statusie dla każdego z modułów określonych na liście w bloku parametrów, a następnie zapisuje te dane do bloku parametrów. Blok ten wymaga (N+4) słów pamięci zmiennych, gdzie N jest liczbą modułów, do których zapisywane będą dane.

Zamieszczona na następnej stronie tabela jest pomocna przy interpretowaniu parametrów wyjściowych:

Tabela 12-5. Wartości parametrów wyjściowych dla funkcji odczytu dodatkowych danych

Pozycja	Pole	Znaczenie
Adres	Funkcja	1 = odczyt dodatkowych danych o statusie
Adres +1	Kod błędu	Jeżeli stwierdzony zostanie brak modułu lub jeżeli jest on nieodpowiedni, albo nie pracuje, funkcja umieszcza kod błędu. Szczegółowe informacje podano w punkcie "Kody błędów", na stronie 12-69.
Adres +2	Kaseta & gniazdo dla błędu	Numer kasety i gniazda, określający miejsce wystąpienia błędu.
Adres +3	Pierwsza kasetka & gniazdo	Numer kasety i gniazda (w postaci liczby heksadecymalnej KKG, gdzie KK jest numerem kasety a G jest numerem gniazda) 1-ego modułu, z którego odczytywane są dane
Adres +4	Odczyt danych z pierwszego modułu	W tym miejscu umieszczane są dane odczytane z pierwszego modułu
Adres +5	Druga kasetka & gniazdo	Numer kasety i gniazda (w postaci liczby heksadecymalnej KKG, gdzie KK jest numerem kasety a G jest numerem gniazda) 2-ego modułu, z którego odczytywane są dane
Adres +6	Odczyt danych z drugiego modułu	W tym miejscu umieszczane są dane odczytane z drugiego modułu
Adres + (I*2) + 1	i-ta kasetka & gniazdo	Numer kasety i gniazda (w postaci liczby heksadecymalnej KKG, gdzie KK jest numerem kasety a G jest numerem gniazda) i-ego modułu, z którego odczytywane są dane
Adres + (I*2) + 2	Odczyt danych z I-ego modułu	W tym miejscu umieszczane są dane odczytane z I-ego modułu
Adres + (N*2) + 1	Ostatnia kasetka & gniazdo	Numer kasety i gniazda (w postaci liczby heksadecymalnej KKG, gdzie KK jest numerem kasety a G jest numerem gniazda) ostatniego modułu, z którego odczytywane będą dane
Adres + (N*2) + 2	Odczyt danych z ostatniego modułu	W tym miejscu umieszczane są dane odczytane z ostatniego modułu
Adres + (N*2) + 3	Wskaźnik końca listy	Umieszczenie wartości zero informuje o zakończeniu listy modułów

Zapis danych (Funkcja #2)

Funkcja ta zapisuje wartość z zakresu 0-15 z bloku parametrów do jednego lub więcej modułów, podanych na liście w bloku parametrów. Blok ten wymaga (N+4) słów pamięci zmiennych, gdzie N jest liczbą modułów, do których zapisywane będą dane.

Tabela 12-6. Wartości parametrów wyjściowych dla funkcji zapisu dodatkowych danych

Pozycja	Pole	Znaczenie
Adres	Funkcja	2 = Zapis danych
Adres +1	Kod błędu	Jeżeli stwierdzony zostanie brak modułu lub jeżeli jest on nieodpowiedni, albo nie pracuje, funkcja umieszcza kod błędu. Jeżeli funkcja zostanie wykonana bez błędów, ale dane nie zostaną poprawnie zapisane do któregośkolwiek z modułów, nie następuje zapisanie kodu błędu. Szczegółowe informacje podano w punkcie "Kody błędów", na stronie 12-69.
Adres +2	Kaseta & gniazdo dla błędu	Numer kasety i gniazda, określający miejsce wystąpienia błędu.
Adres +3	Pierwsza kasetka & gniazdo	Numer kasety i gniazda (w postaci liczby heksadecymalnej KKG, gdzie KK jest numerem kasety a G jest numerem gniazda) 1-ego modułu, do którego zapisywane będą dane
Adres +4	Zapis danych do pierwszego modułu	Zapis danych do pierwszego modułu
Adres +5	Druga kasetka & gniazdo	Numer kasety i gniazda (w postaci liczby heksadecymalnej KKG, gdzie KK jest numerem kasety a G jest numerem gniazda) 2-ego modułu, do którego zapisywane będą dane
Adres +6	Zapis danych do drugiego modułu	Zapis danych do drugiego modułu
Adres + (I*2) + 1	i-ta kasetka & gniazdo	Numer kasety i gniazda (w postaci liczby heksadecymalnej KKG, gdzie KK jest numerem kasety a G jest numerem gniazda) I-ego modułu, do którego zapisywane będą dane.
Adres + (I*2) + 2	Zapis danych do I-ego modułu	Zapis danych do I-ego modułu.
Adres + (N*2) + 1	Ostatnia kasetka & gniazdo	Numer kasety i gniazda (w postaci liczby heksadecymalnej KKG, gdzie KK jest numerem kasety a G jest numerem gniazda) ostatniego modułu, do którego zapisywane będą dane.
Adres + (N*2) + 2	Zapis danych do ostatniego modułu	Zapis danych do ostatniego modułu
Adres + (N*2) + 3	Wskaźnik końca listy	Umieszczenie wartości zero informuje o zakończeniu listy modułów

Odczyt/ zapis danych (Funkcja #3)

Funkcja ta czyta słowo dodatkowych danych o statusie z modułu określonego w bloku parametrów, a następnie zapisuje wartość z zakresu 0 - 15 z bloku parametrów do tego modułu. Proces odczytu i zapisu powtarzany jest dla każdego z modułów określonych na liście w bloku parametrów. Blok ten wymaga $(N*4)+3$ słów pamięci zmiennych, gdzie N jest liczbą modułów, z którymi wymieniane będą dane.

Tabela 12-7. Wartości parametrów wyjściowych dla funkcji zapisu/ odczytu dodatkowych danych

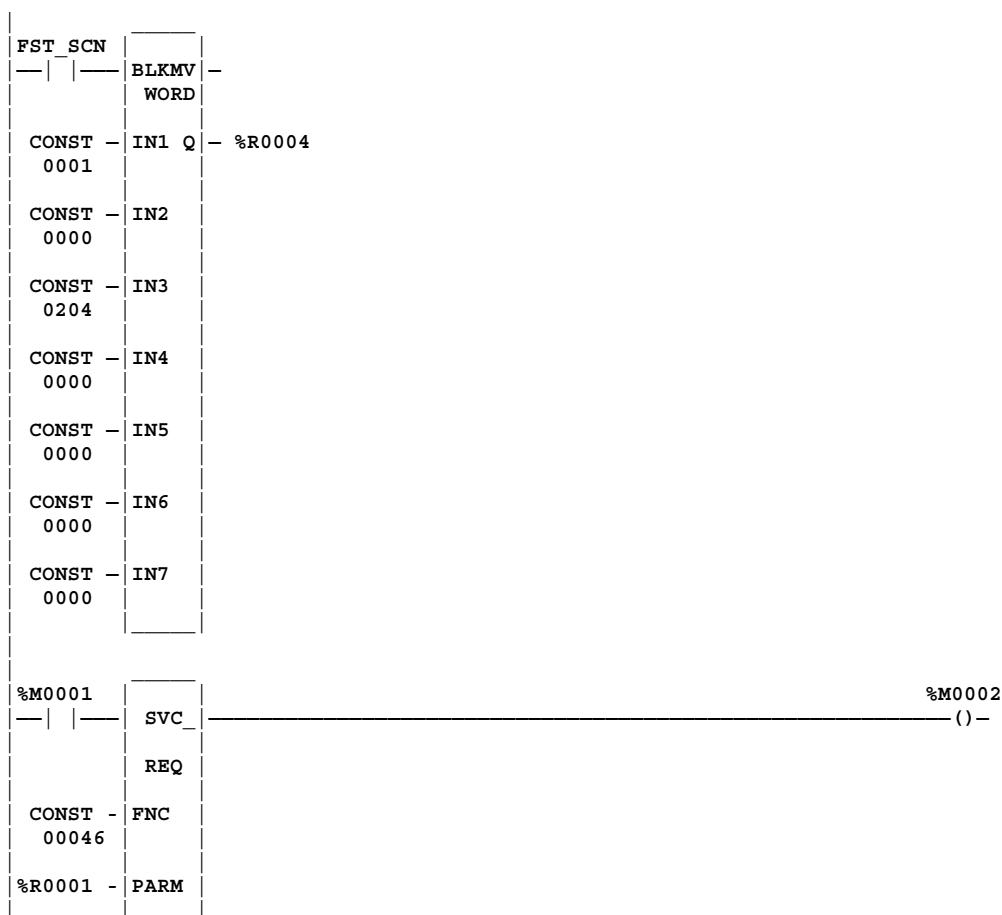
Pozycja	Pole	Znaczenie
Adres	sekwencja	3 = odczyt/ zapis
Adres +1	Kod błędu	Jeżeli stwierdzony zostanie brak modułu lub jeżeli jest on nieodpowiedni, albo nie pracuje, funkcja umieszcza kod błędu. Jeżeli funkcja zostanie wykonana bez błędów, ale dane nie zostaną poprawnie zapisane do któregośkolwiek z modułów, nie następuje zapisanie kodu błędu. Szczegółowe informacje podano w punkcie "Kody błędów", na stronie 12-69.
Adres +2	Kaseta & gniazdo dla błędu	Numer kasety i gniazda, określający miejsce wystąpienia błędu.
Adres +3	Pierwsza kasetka & gniazdo	Numer kasety i gniazda (w postaci liczby heksadecymalnej KKG, gdzie KK jest numerem kasety a GG jest numerem gniazda) 1-go modułu, z którym wymieniane będą dane
Adres +4	Odczyt danych z pierwszego modułu	W tym miejscu umieszczane są dane odczytane z pierwszego modułu
Adres +5	Zapis danych do pierwszego modułu	Zapis danych do pierwszego modułu
Adres +6	Druga kasetka & gniazdo	Numer kasety i gniazda (w postaci liczby heksadecymalnej KKG, gdzie KK jest numerem kasety a GG jest numerem gniazda) 2-go modułu, z którym wymieniane będą dane
Adres +7	Odczyt danych z drugiego modułu	W tym miejscu umieszczane są dane odczytane z drugiego modułu
Adres +8	Zapis danych do drugiego modułu	Zapis danych do drugiego modułu
Adres + $((I-1) * 3) + 3$	i-ta kasetka & gniazdo	Numer kasety i gniazda (w postaci liczby heksadecymalnej KKG, gdzie KK jest numerem kasety a GG jest numerem gniazda) I-ego modułu, z którym wymieniane będą dane
Adres + $((I-1) * 3) + 4$	Odczyt danych z I-ego modułu	W tym miejscu umieszczane są dane odczytane z I-ego modułu
Adres + $((I-1) * 3) + 5$	Zapis danych do I-ego modułu	Zapis danych do I-ego modułu
Adres + $((N-1) * 3) + 3$	Ostatnia kasetka & gniazdo	Numer kasety i gniazda (w postaci liczby heksadecymalnej KKG, gdzie KK jest numerem kasety a GG jest numerem gniazda) ostatniego modułu, z którym wymieniane będą dane
Adres + $((N-1) * 3) + 4$	Odczyt danych z ostatniego modułu	W tym miejscu umieszczane są dane odczytane z ostatniego modułu
Adres + $((N-1) * 3) + 5$	Zapis danych do ostatniego modułu	Zapis danych do ostatniego modułu
Adres + $(N*3) + 3$	Wskaźnik końca listy	Umieszczenie wartości zero informuje o zakończeniu listy modułów

Kody błędów

Wartość	Opis
1	Pomyślne zakończenie wykonywania bloku funkcyjnego.
-1	Brak modułu w określonym gnieździe.
-2	Niewłaściwy moduł - moduł w podanym gnieździe nie jest modułem wyspecjalizowanym lub nie obsługuje tej funkcjonalności.
-3	Moduł nie pracuje - moduł w podanym gnieździe nie komunikuje się w poprawny sposób z jednostką centralną.
-4	Błąd parzystości przy odczycie - przy odczycie danych z kasety rozszerzającej lub zdalnej wystąpił błąd parzystości.
-5	W bloku parametrów podano niepoprawną funkcję.

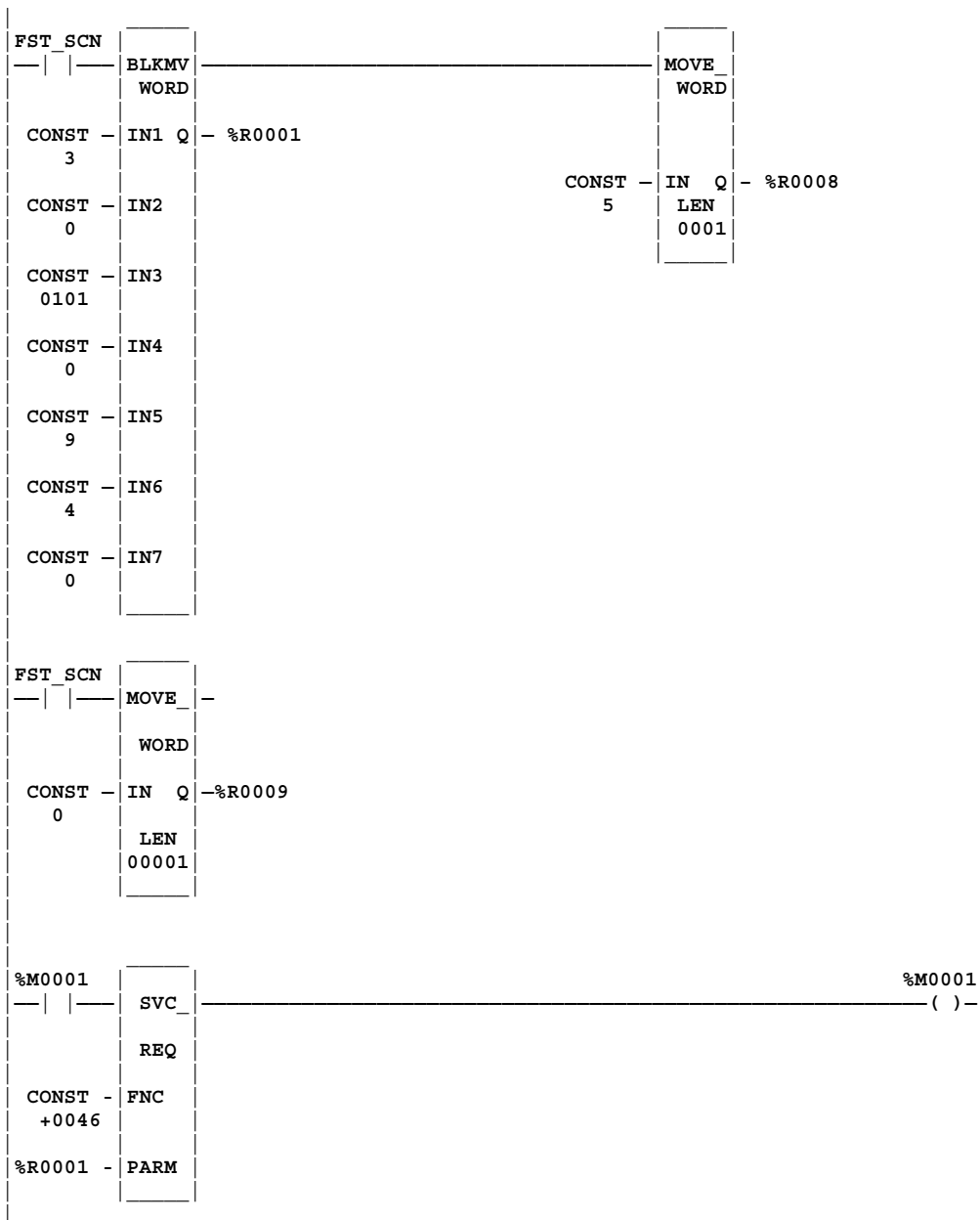
Przykład 1

W zamieszczonym poniżej przykładzie odczytywane są dane dla jednego modułu, zainstalowanego w kasecie 2, gnieździe 4. Parametry IN4 do IN7 muszą być równe zero (0). Parametry IN6 do IN7 nie mają zastosowania w niniejszym przykładzie. Po pomyślnym wykonaniu bloku funkcyjnego, dane zapisywane są do %R0004.



Przykład 2

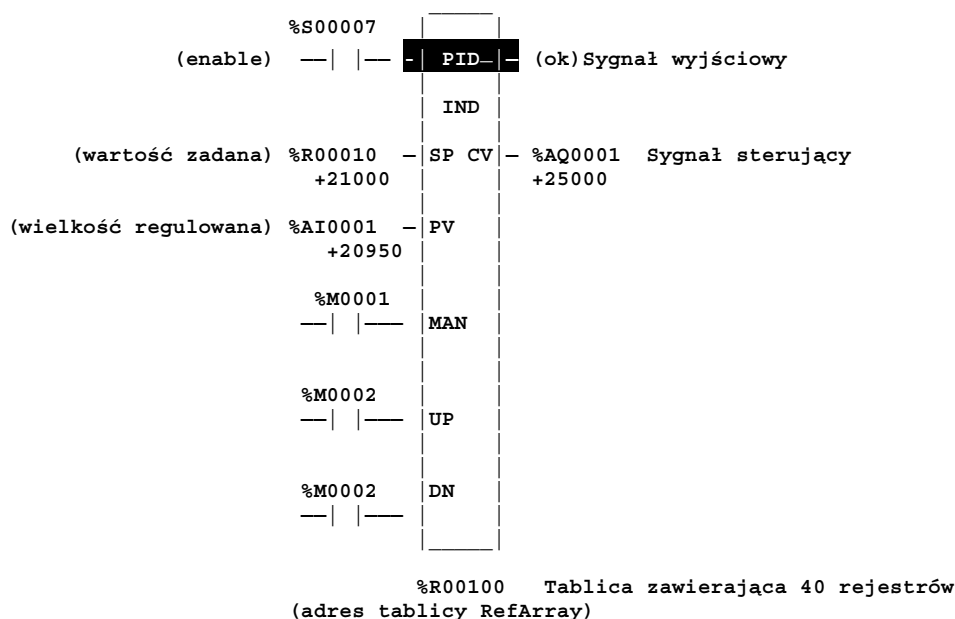
W przykładzie tym odczytywane są dodatkowe dane o statusie z modułu zainstalowanego w kasecie 0, gnieździe 4 oraz z modułu zainstalowanego w kasecie 1, gnieździe 1. Do modułu pierwszego zapisywana jest wartość 5 a do drugiego 9. Warto zwrócić uwagę, że moduły nie muszą być uszeregowane na liście zgodnie z numerami gniazd. Dane odczytane z modułu zainstalowanego w kasecie 0, gnieździe 4 zostaną przekazane za pomocą zmiennej %R0007. Dane odczytane z modułu zainstalowanego w kasecie 1, gnieździe 1 zostaną przekazane za pomocą zmiennej %R0004.



PID

Blok proporcjonalno- całkowo- różniczkowy PID rozwiązuje algorytm sterowania zamkniętego układu regulacji. Blok ten na podstawie dwóch wartości: punktu ustalonego (wartości zadanej) i bieżącej wartości wielkości regulowanej oraz korzystając z zadanych parametrów bloku, oblicza taką wartość sygnału sterującego (nastawiającego), która spowoduje zmniejszenie uchybu, czyli odchylenia wielkości regulowanej od wartości zadanej (punktu pracy).

Blok funkcyjny PID wykorzystuje 40 rejestrów w pamięci sterownika, w których przechowuje zbiór parametrów regulatora. Wszystkie parametry są 16 bitowymi liczbami całkowitymi, co ma na celu zapewnienie kompatybilności z 16 bitowymi analogowymi wielkościami regulowanymi. Pozwala to na zastosowanie dla wartości regulowanych PV pamięci adresowanej przez %AI oraz pamięci adresowanej przez %AQ dla sygnału sterującego. Rysunek zamieszczony poniżej obrazuje typowe parametry bloku funkcyjnego PID:



Ponieważ wiele parametrów jest skalowanymi liczbami całkowitymi, 16 bitowymi, ich wartość musi być podawana w jednostkach PV lub w jednostkach bezwymiarowych PV, albo w jednostkach CV lub jednostkach bezwymiarowych CV. Przykładowo, parametr SP musi być przeskalowany do takiego samego zakresu jak parametr PV, ponieważ blok PID oblicza uchyb jako różnicę pomiędzy wartościami tych parametrów. Parametry wielkość regulowana PV i wielkość ustawiająca CV mogą przyjmować wartości z zakresu -32000 lub 0 do 32000 z dopasowaniem do skalowania wielkości analogowych lub z zakresu 0 do 10000 co pozwala na wyświetlanie wartości w formacie 0.00% do 100.00%. Wartości parametrów PV i CV nie muszą być tak samo skalowane, współczynniki skalowania są wtedy zawarte we współczynnikach wzmocnienia regulatora PID.

Uwaga

Blok PID nie może być wykonywany częściej niż co 10 milisekund. Jeżeli więc blok ten ma być wykonywany w każdym cyklu, a czas trwania cyklu jest krótszy od 10 milisekund, może to być powodem nie uzyskiwania żądanych wyników. W takim przypadku, blok PID zostanie wywołany dopiero po kilku cyklach, dokładnie po upływie 10 milisekund. Przykładowo, jeżeli cykl pracy jednostki centralnej sterownika wynosi 9 milisekund, blok funkcyjny jest wykonywany co drugi cykl pracy jednostki centralnej, czyli co 18 ms.

Parametry

Parametr	Opis
enable	Sygnal wejściowy.
SP	Wartość zadana wielkości regulowanej (punkt pracy regulatora). Po porównaniu parametrów wielkości regulowanej PV z wielkością zadaną SP blok PID dobiera tak sygnał sterujący CV, aby parametry PV i SP miały taką samą wartość (uchyb zerowy).
PV	Wielkość regulowana, której wartość odczytywana jest ze sterowanego procesu, deklarowana zwykle jako zmienna typu %AI.
MAN	Parametr MAN o wartości 1 powoduje przełączenie regulatora w tryb sterowania ręcznego (MANUAL). Wartość tego parametru równa 0 powoduje przełączenie regulatora w tryb sterowania automatycznego.
UP	Parametr mający znaczenie tylko w trybie sterowania ręcznego*. Wartość tego parametru równa 1 powoduje zwiększenie wartości sygnału nastawiającego, wartość 0 nie wywołuje żadnego działania.
DN	Parametr mający znaczenie tylko w trybie sterowania ręcznego*. Wartość tego parametru równa 1 powoduje zmniejszenie wartości sygnału ustawiającego, wartość 0 nie wywołuje żadnego działania.
Adres tablicy RefArray	Adres pierwszego z rejestrów, w których przechowywane są wewnętrzne parametry regulatora (parametry użytkownika i parametry wewnętrzne). Obszar ten zajmuje 40 rejestrów pamięci typu %R, które nie mogą być wykorzystywane w innym celu.
ok	Sygnal wyjściowy, wysyłany po poprawnym wykonaniu bloku funkcyjnego.
CV	Sygnal sterujący (ustawiający) procesu, często jest to zmienna typu %AQ.

* Inkrementowany (parametr UP) lub dekrementowany (parametr DN) o jeden (1) w jednym wywołaniu bloku funkcyjnego PID.

Dopuszczalne typy parametrów

Parametr	Sygnal	%I	%Q	%M	%T	%S	%G	%R	%AI	%AQ	Stała	Brak
enable	•											
SP		•	•	•	•		•	•	•	•	•	
PV		•	•	•	•		•	•	•	•		
MAN	•											
UP	•											
DN	•											
adres								•				
ok	•											•
CV		•	•	•	•		•	•	•	•		

• Dozwolony typ parametru.

Blok parametrów

Poza 2 parametrami wejściowymi i 3 ręcznie sterowanymi stykami, blok PID korzysta z 13 parametrów zapisanych w tablicy RefArray. Wartość tych parametrów należy ustawić przed wywołaniem tego bloku. Pozostałe parametry są wykorzystywane przez sterownik i nie mogą być zmieniane. Zmienna %Ref w zamieszczonej poniżej tabeli jest adresem tablicy RefArray, pokazanej na schemacie bloku funkcyjnego PID. Liczba podana po znaku plus określa pozycję w tabeli. Przykładowo, jeżeli adres początkowy tabeli RefArray jest równy %R100, zmienna %R113 zawiera wartość sygnału sterującego w ręcznym trybie sterowania.

Tabela 12-8. Zestawienie parametrów bloku funkcyjnego PID

Rejestr	Parametr	Jednostki program.	Zakres wartości
%Ref+0000	Loop Number (Numer układu regulacji)	Integer	0 do 255 (wyłącznie na potrzeby informowania użytkownika).
%Ref+0001	Algorithm (Rodzaj algorytmu)	Wartość ustawiana przez sterownik, tylko do odczytu	Nie konfigurow.
%Ref+0002	Sample Period (Okres próbkowania)	10 milisekund	0 (każdy cykl) do 65535 (10.9 min.). Dla sterowników serii 90-30 należy wprowadzić wartość co najmniej 10 (porównać z uwaga na stronie 12-71).
%Ref+0003	Dead Band (Górna granica strefy nieczułości)	Jednostki bezwym. PV	0 do 32000 (wartości ujemne nie dozwolone).
%Ref+0004	Dead Band —(Dolna granica strefy nieczułości)	Jednostki bezwym. PV	-32000 do 0 (wartości dodatnie nie dozwolone).
%Ref+0005	Proportional Gain—Kp (Współczynnik wzmocnienia proporcjonalnego Kp)	0.01 CV%/PV%	0 do 327.67 %/%
%Ref+0006	Derivative Gain—Kd (Współczynnik wzmocnienia różniczkowego Kd)	0.01 sekundy	0 do 327.67 sek.
%Ref+0007	Integral Rate—Ki (Współczynnik wzmocnienia całkowego Ki)	powtórzeń/ 1000 sek.	0 do 32.767 powtórzeń/ sek.
%Ref+0008	CV Bias/Output Offset (Przesunięcie punktu pracy)	Jednostki bezwym. parametru CV	-32000 do 32000 (dodawana do wyjścia bloku całkowego).
%Ref+0009	Upper Clamp ((Górna granica wartości sygnału ustawiającego)	Jednostki bezwym. parametru CV	-32000 to 32000 (>%Ref+10) wartości granicznych sygnału wyjściowego.
%Ref+0010	Lower Clamp (Dolna granica wartości sygnału sterującego)	Jednostki bezwym. parametru CV	-32000 to 32000 (<%Ref+09) wartości granicznych sygnału wyjściowego.
%Ref+0011	Minimum Slew Time (Minimalny czas narastania sygnału ustawiającego)	Sekund/ pełne przemieszczenie	0 (brak) do 32000 sek. do przesunięcia o 32000 CV.
%Ref+0012	Config Word- Parametr konfiguracyjny	5 młodszych bitów	Bity 0 do 2 informują o sposobie obliczania uchybu, polaryzacji sygnału ustawiającego CV i sposobie różniczkowania.
%Ref+0013	Manual Command (Sygnał sterujący w trybie ręcznym)	Jednostki bezwym. parametru CV	Nadaża za parametrem CV w trybie automatycznym lub określa wartość CV w trybie ręcznym.
%Ref+0014	Control Word ((Słowo sterujące)	Zapisywany przez sterownik, o ile bit 1 <> 1	Jeżeli wartość bitu 1 wynosi 0, wartość ta zapisywana jest przez sterownik. Proszę porównać z punktem "Szczegółowy opis parametrów bloku funkcyjnego PID" na stronie 12-76.
%Ref+0015	Internal SP (Rejestr wewnętrzny wielkości zadanej SP)	Wartość ustawiana przez sterownik, tylko do odczytu	Nie konfigurow.
%Ref+0016	Internal CV (Rejestr wewnętrzny wielkości ustawiającej CV)	Wartość ustawiana przez sterownik, tylko do odczytu	Nie konfigurow.

Rejestr	Parametr	Jednostki program.	Zakres wartości
%Ref+0017	Internal PV (Rejestr wewnętrzny wielkości regulowanej PV)	Wartość ustawiana przez sterownik, tylko do odczytu	Nie konfigurow.
%Ref+0018	wyjść	Wartość ustawiana przez sterownik, tylko do odczytu	Nie konfigurow.
%Ref+0019	Diff Term Storage (Dane robocze dotyczące bloku różniczkującego)	Wartość ustawiana przez sterownik, tylko do odczytu	Nie konfigurow.
%Ref+0020 i %Ref+0021	Int Term Storage ((Dane robocze dotyczące bloku całkującego)	Wartość ustawiana przez sterownik, tylko do odczytu	Nie konfigurow.
%Ref+0022	Slew Term Storage (Dane robocze dla prędkości narastania sygnału ustawiającego)	Wartość ustawiana przez sterownik, tylko do odczytu	Nie konfigurow.
%Ref+0023	Clock (Zegar wewnętrzny, czas który upłynął od momentu ostatniego wykonania algorytmu)	Wartość ustawiana przez sterownik, tylko do odczytu	Nie konfigurow.
%Ref+0024			
%Ref+0025			
%Ref+0026	Y Remainder Storage (Rejestr do przechowywania wartości Y - wewnętrzna zmienna sterownika)	Wartość ustawiana przez sterownik, tylko do odczytu	Nie konfigurow.
%Ref+0027	Lower Range for SP, PV (Dolna granica zakresu wartości parametrów SP i PV)	Jednostki bezwym. PV	-32000 do 32000 (>%Ref+28) do wyświetlania
%Ref+0028	Upper Range for SP, PV (Górna granica zakresu wartości parametrów SP i PV)	Jednostki bezwym. PV	-32000 do 32000 (do wyświetlania
%Ref+0029 ...	Zarezerwowane do użytku wewnętrznego	brak	Nie konfigurow.
%Ref+0034			
%Ref+0035 ...	Zarezerwowane do użytku wewnętrznego	brak	Nie konfigurow.
%Ref+0039			

W sterownikach 90-30 tablica RefArray musi być przechowywana w obszarze pamięci adresowanej przez %R. Należy zwrócić uwagę, że każdy blok funkcyjny PID musi korzystać z innego obszaru pamięci, nawet jeżeli wszystkich 13 parametrów jest takich samych. Jest to spowodowane przez fakt, że pozostałe rejestry tablicy wykorzystywane są do pamiętania danych wewnętrznych bloku funkcyjnego PID. Należy upewnić się, że tablica mieści się w całości w pamięci.

W celu skonfigurowania parametrów należy zaznaczyć blok funkcyjny PID, a następnie wcisnąć **F10** w celu wyświetlenia ekranu zawierającego parametry, po czym za pomocą klawiszy kursora przejść do odpowiednich pól i wprowadzić żądane wartości. Jako wartość domyślną można w większości przypadków wprowadzić 0, za wyjątkiem parametru Upper Clamp, który musi być większy od parametru Lower Clamp, co jest warunkiem wykonania bloku PID. Należy zwrócić uwagę, że blok PID **nie** przesyła sygnału wyjściowego w przypadku wykrycia błędu w parametrach użytkownika, przez co, w czasie modyfikowania wartości, zalecane jest monitorowanie jego statusu za pomocą chwilowego przekaźnika (%T).

Po ustawieniu odpowiednich wartości bloku funkcyjnego PID powinny one zostać zdefiniowane jako stałe za pomocą funkcji BLKMOV, dzięki czemu będzie można jest powtórnie załadować jeżeli wymagane będzie ponowne wprowadzenie domyślnych wartości użytkownika dla tego bloku.

Opis działania bloku funkcyjnego PID

Jeżeli do wejścia "enable" bloku funkcyjnego doprowadzony zostanie sygnał wejściowy, a jednocześnie do wejścia MAN sygnał nie jest doprowadzany (wartość parametru MAN wynosi 0), realizowany jest algorytm PID. Czas, który upłynął od ostatniego wykonania bloku PID porównywany jest z zaprogramowanym okresem próbkowania (%Ref + 2). Jeśli czas ten osiągnął wartość większą lub równą okresowi próbkowania, wykonywany jest algorytm PID (rozwiązywane jest równanie zamkniętego układu regulacji), dla którego podstawą czasu jest okres, jaki upłynął od ostatniego wykonania bloku PID, a nie zaprogramowany okres próbkowania. W trybie automatycznym, wartość sygnału sterującego przypisywana jest do parametru Sygnał sterujący w trybie ręcznym o adresie %Ref + 13.

Jeżeli blok funkcyjny PID pracuje w trybie sterowania ręcznego (do wejścia MAN dopływa sygnał), wartość sygnału ustawiającego jest odczytywana z rejestru o adresie "%Ref + 13" i może być zwiększana lub zmniejszana za pomocą parametrów wejściowych UP i DN bloku. W celu szybkiej zmiany wyjściowego sygnału ustawiającego, możliwe jest również dodanie lub odjęcie dowolnej wartości w jednostkach sygnału ustawiającego bezpośrednio do/ od parametru Sygnał sterujący w trybie ręcznym (%Ref + 13).

Wartość sygnału ustawiającego CV ograniczana jest w bloku PID za pomocą parametrów: CV Upper Clamp i CV Lower Clamp. W przypadku wprowadzenia dodatkowej wartości parametru Minimum Slew Time, wartość ta jest wykorzystywana do ograniczenia prędkości zmian wartości sygnału ustawiającego CV.

W przypadku przekroczenia amplitudy lub prędkości zmian sygnału ustawiającego CV, układ całkujący ustawi wartość graniczną tego parametru. W wyniku działania mechanizmu zapobiegającego przekroczeniu dopuszczalnych wartości (opisanego na stronie 12-78), jeżeli wartość sygnału ustawiającego CV jest większa (lub mniejsza) od wartości granicznych przez dłuższy okres czasu, zmiana znaku błędu powoduje szybkie odejście od wartości granicznej.

Mechanizm taki, w połączeniu z parametrem Manual Command nadążającym za parametrem CV w trybie automatycznego sterowania zapewnia możliwość płynnego przejścia pomiędzy trybami sterowania automatycznego a ręcznego. Parametry CV Upper Clamp i CV Lower Clamp oraz Minimum Slew Time są nadal wykorzystywane w trybie ręcznym, a parametry wewnętrzne zapamiętane w bloku całkującym są uaktualniane. Oznacza to, że przejście do trybu ręcznego sterowania nie spowoduje przekroczenia maksymalnej prędkości narastania sygnału ustawiającego, określonej przy użyciu Minimum Slew Time oraz nie przekroczone zostaną wartości graniczne, określone parametrami CV Upper Clamp i CV Lower Clamp.

Uwaga

Ten sam blok funkcyjny PID nie może być wywoływany więcej niż jeden raz w ciągu jednego cyklu.

Zamieszczona poniżej tabela zawiera szczegółowe informacje o parametrach opisanych wstępnie w Tabeli 12-3. Liczby podane w nawiasach, po nazwie parametru, określają pozycję w tabeli RefArray.

Tabela 12-9. Szczegółowy opis parametrów bloku funkcyjnego PID

Parametr	Opis
Loop Number - Numer układu regulacji (00)	Liczba całkowita bez znaku, która pozwala na identyfikację układu regulacji poprzez jego numer. Numer ten wyświetlany jest pod adresem bloku parametrów podczas śledzenia wykonywania programu sterującego przez sterownik za pomocą oprogramowania Logicmaster 90-30/20/Micro. Jest to parametr opcjonalny.
Algorithm - Rodzaj algorytmu PID (01)	Liczba całkowita bez znaku, której wartość ustawiana jest przez sterownik, w celu określenia, który z dwóch dostępnych algorytmów PID jest realizowany. Wartość równa 1 oznacza standardowy algorytm ISA, natomiast wartość 2 oznacza algorytm o niezależnych wyrazach.
Sample Period - Okres Próbki (impulsowania) (02)	<p>Odstęp czasowy (mierzony w setnych częściach sekundy) pomiędzy dwoma kolejnymi wykonaniami bloków funkcyjnych PID. Przykładowo, jeżeli okres ten ma wynosić 100 milisekund, należy wprowadzić wartość 10. Wprowadzenie wartości 0 powoduje, że blok funkcyjny jest realizowany przy każdym doprowadzeniu sygnału sterującego.</p> <p>Algorytm PID realizowany jest wyłącznie pod warunkiem, że czas wskazywany przez zegar sterownika jest równy lub większy od czasu poprzedniej realizacji tego algorytmu + Okres próbkowania. Należy pamiętać, że sterowniki 90-30 nie pozwalają na wprowadzenie okresu próbkowania o wartości mniejszej od 10 milisekund (porównać z Uwagą na stronie 12-71), w przypadku krótszych czasów realizacji cyklu, bloki funkcyjne PID będą ignorowane. Sterownik zapamiętuje czas, który upłynął od ostatniej realizacji algorytmu PID z dokładnością do 100 mikrosekund. Jeżeli wartość tego parametru wynosi 0, algorytm PID jest realizowany za każdym razem, gdy do bloku funkcyjnego doprowadzony zostanie sygnał wyjściowy.</p>
Dead Band (+/-) Górna i dolna granica strefy nieczułości (03/04)	Wartości całkowite ze znakiem, określające górną i dolną granice strefy nieczułości w jednostkach bezwymiarowych PV. Jeżeli strefa nieczułości nie jest wymagana, wartości tych parametrów powinny być ustawione na 0. Jeżeli uchyb wielkości regulowanej (czyli różnica pomiędzy wartością zadaną a wartością bieżącą (SP-PV) lub różnica pomiędzy wartością bieżącą a wartością zadaną (PV-SP) mieszczą się w przedziale określonym przez górną i dolną granicę strefy nieczułości, algorytm PID realizowany jest przy założeniu, że uchyb jest równy 0. Jeżeli uchyb nie ma wartości zerowej, górna granica strefy nieczułości musi być większa od zera (0), a dolna mniejsza od zera (0), bowiem w przeciwnym wypadku blok funkcyjny PID nie zostanie wykonany. <i>Należy pozostawić wartość 0 tych parametrów do momentu ustawienia i dostrojenia regulatora.</i> Później można wprowadzić inne wartości tych parametrów w celu uniknięcia małych zmian wyjściowego sygnału ustawiającego CV, wynikłych z małych wartości błędów, mogących być powodem ewentualnego przyspieszonego zużycia mechanicznego członu wykonawczego.
Proportional Gain-Kp – Współczynnik proporcjonalnego Kp (05)	Wartość całkowita, oznaczana symbolem Kc w wersji ISA, określająca zmianę sygnału ustawiającego CV przy zmianie uchybu o 100 jednostek bezwymiarowych PV. Parametr ten wyświetlany jest jako 0.00 %/%, z domyślnie przyjmowanymi dwoma miejscami dziesiętnymi. Przykładowo parametr Kp równy 450 będzie wyświetlany jako 4.50, a powodowana przez niego zmiana sygnału nastawiającego będzie wynosić $Kp \cdot Uchyb / 100$ lub $450 \cdot Uchyb / 100$. Ogólnie ujmując, jest to pierwsza nastawa zmieniana przy strojeniu regulatora.
Derivative Gain-Kd - Czas różniczkowania Kd (06)	Wartość całkowita określająca czas różniczkowania w setnych częściach sekundy. Wprowadzana w jednostkach 10 milisekund, a wyświetlana w formacie 0.00 sekund, z 2 miejscami na część dziesiętną. Przykładowo, jeżeli parametr Kd ma wartość 120, będzie on wyświetlany jako 1.20 sek., a powodowana przez niego zmiana sygnału wyjściowego bloku PID będzie wynosić $Kd \cdot \text{przyrost uchybu} / \text{przyrost czasu}$, czyli $120 \cdot 4 / 3$, jeżeli Uchyb zmieni się o 4 jednostki bezwymiarowe PV w ciągu 30 milisekund. Parametr KD może zostać wykorzystany do przyspieszenia wolnej odpowiedzi regulatora, ale jest bardzo wrażliwy na zakłócenia wielkości regulowanej PV.
Integral Rate Gain-Ki - Współczynnik wzmocnienia członu całkującego - Ki (07)	Wartość całkowita ze znakiem określająca częstotliwość całkowania, jeżeli Uchyb ma stałą wartość równą 1 jednostce bezwymiarowej PV. Parametr ten wyświetlany jest w formacie 0.000 powtórzeń/sek, z przeznaczeniem 3 miejsc na część dziesiętną. Przykładowo, jeżeli parametr Ki ma wartość 1400, wyświetlany będzie jako 1.400 powtórzeń/sek., a powodowana przez niego zmiana sygnału wyjściowego bloku PID wynosi $Ki \cdot Uchyb \cdot dt$, czyli $1.400 \cdot 20 \cdot 50 / 1000$, dla Uchybu o wartości 20 jednostek bezwymiarowych wartości regulowanej PV i czasu trwania cyklu sterownika 50 milisekund (Okres próbkowania równy 0). Parametr Ki jest zwykle drugą nastawą ustawianą przy strojeniu regulatora, po parametrze Kp.
CV Bias/Output Offset - Przesunięcie punktu pracy (08)	Wartość całkowita w jednostkach bezwymiarowych CV, dodawana do sygnału wyjściowego bloku PID, przed zastosowaniem wartości granicznych prędkości zmian i amplitudy. Parametr ten można wykorzystać do ustawienia nie zerowej wartości sygnału ustawiającego CV, jeżeli stosowany jest wyłącznie współczynnik wzmocnienia proporcjonalnego lub do sterowania z oddziaływaniem ("feed forward control").

Tabela 12-9. Zestawienie parametrów bloku funkcyjnego PID (kontynuacja)

Parametr	Opis
CV Upper and Lower Clamps – Górna i dolna granica wartości sygnału ustawiającego (09/10)	Wartości całkowite ze znakiem, określające największą i najmniejszą wartość sygnału ustawiającego. Jeżeli wartości tych parametrów nie zostaną wprowadzone lub jeżeli wartość granicy górnej jest mniejsza od granicy dolnej, blok funkcyjny PID nie zostanie wykonany. Wartości tych granic są zwykle wyznaczane w oparciu o ograniczenia fizyczne sygnału ustawiającego CV. Są one również wykorzystywane do graficznej reprezentacji sygnału ustawiającego CV, przy użyciu oprogramowania LM90. Blok funkcyjny PID monitoruje wartość wewnętrzną bloku całkującego, zabezpieczając tym samym przed przekroczeniem dopuszczalnych wartości.
Minimum Slew Time – Minimalny czas narastania sygnału ustawiającego (11)	Wartość całkowita dodatnia, definiująca minimalną liczbę sekund, wymaganą do przejścia sygnału ustawiającego od wartości 0 do 100% lub 32000 jednostek bezwymiarowych CV. Jest to odwrotność dopuszczalnej prędkości zmian sygnału ustawiającego CV. Jeżeli parametr ten ma wartość dodatnią, wielkość zmiany sygnału CV w jednostkach bezwymiarowych CV nie może przekroczyć 32000 razy przyrost czasu (w sekundach) podzielonego przez minimalny czas narastania sygnału ustawiającego. Przykładowo, jeżeli okres próbkowania wynosi 2.5 sekundy, a minimalny czas narastania sygnału ustawiającego jest równy 500 sekund, sygnał ustawiający CV nie może się zmienić o więcej niż $32000 \cdot 2.5 / 500$, czyli o 160 jednostek bezwymiarowych sygnału CV w ciągu jednego obliczania bloku PID. Podobnie jak w przypadku wartości granicznych sygnału ustawiającego, dostępny jest mechanizm zapobiegający przekroczeniu dopuszczalnych wartości, zmieniający wartość bloku całkującego w przypadku przekroczenia wartości granicznych sygnału ustawiającego CV. Jeżeli Minimalny czas narastania sygnału ustawiającego jest równy 0, nie ma żadnego ograniczenia co do prędkości narastania. Wartość 0 należy wprowadzać w czasie dostrajania bloku PID.
Config Word – Parametr konfiguracyjny (12)	<p>5 bitów tego słowa wykorzystywanych jest do modyfikowania trzech standardowych parametrów regulatora PID. Wartość pozostałych bitów powinna być ustawiona na 0. Bit zerowy modyfikuje sposób obliczania uchybu ze standardowego (SP-PV) na (PV-SP), a więc odwraca znak tego uchybu. Jest to wykorzystywane w układach sterowania, gdzie wzrost wielkości regulowanej powinien powodować spadek sygnału ustawiającego CV. Po ustawieniu na 1 wartości pierwszego bitu, zmieniana jest polaryzacja sygnału ustawiającego CV, parametr ten będzie więc miał wartość ujemną, a nie dodatnią. Ustawienie na 1 wartości trzeciego bitu zmienia działanie mechanizmu zapobiegającego przekraczaniu wartości granicznych.</p> <p>Poniżej przedstawiono znaczenie młodszych 5 bitów parametru konfiguracyjnego:</p> <p>Bit 0 = Sposób obliczania uchybu. Gdy wartość logiczna tego bitu wynosi 0, uchyb obliczany jest według wzoru SP-PV. Gdy wartość logiczna tego bitu wynosi 1, uchyb obliczany jest według zależności PV-SP.</p> <p>Bit 1 = Polaryzacja sygnału ustawiającego CV. Gdy wartość tego bitu wynosi 0, parametr wyjściowy CV stanowi bezpośredni wynik obliczeń algorytmu PID. Gdy wartość tego bitu wynosi 1, parametr wyjściowy CV zawiera zanegowaną wartość wyniku obliczeń algorytmu PID.</p> <p>Bit 2 = Sposób różniczkowania wielkości regulowanej PV. Jeżeli wartość tego bitu wynosi 0, różniczkowany jest uchyb. Jeżeli wartość tego bitu wynosi 1, różniczkowany jest sygnał PV. Wszystkie bity powinny mieć wartość zero.</p> <p>Bit 3 = Działania podejmowane po przekroczeniu strefy nieczułości. Po ustawieniu tego bitu na wartość zero, nie są podejmowane żadne działania. Jeżeli uchyb mieści się w granicach strefy nieczułości, przyjmowana jest zerowa wartość uchybu. W przeciwnym wypadku, wielkość uchybu nie jest modyfikowana przez ograniczenie strefą nieczułości. Po ustawieniu tego bitu na wartość 1, podejmowane są pewne działania związane ze strefą nieczułości. Jeżeli uchyb mieści się w granicach strefy nieczułości, przyjmowana jest zerowa wartość uchybu. Jeżeli jednak uchyb wykracza poza granice strefy nieczułości, wartość uchybu jest zmniejszana o wartości graniczne strefy nieczułości (uchyb = uchyb - wartość graniczna strefy nieczułości).</p> <p>Bit 4 = Mechanizm zapobiegający przekroczeniu wartości granicznych. Po ustawieniu wartości tego bitu na zero, mechanizm zapobiegający przekroczeniu wartości korzysta z algorytmu zerowania wstecz. Jeżeli sygnał wyjściowy mieści się w dopuszczalnych granicach, wartość ta zastępuje zakumulowaną wartość Y (zdefiniowaną na stronie 12-78). bez względu na wartość wymaganą do wygenerowania sygnału wyjściowego mieszczącego się w dopuszczalnym zakresie wartości. Jeżeli wartość tego bitu jest równa jeden, wartość ta zastępuje zakumulowany wyraz Y przez wartość wyrazu Y zapamiętaną w momencie rozpoczęcia obliczeń. Dzięki takiemu rozwiązaniu, wstępna wartość ograniczająca Y jest pamiętana tak długo, jak długo sygnał wyjściowy mieści się w dopuszczalnych granicach.</p> <p>UWAGA: Mechanizm zapobiegający przekroczeniu dopuszczalnych wartości jest dostępny wyłącznie w jednostkach centralnych 90-30 z oprogramowaniem systemowym ver. 6.50 lub nowsza.</p> <p>Przykładowo, po ustawieniu wartości Parametru konfiguracyjnego na 0 w domyślnej konfiguracji regulatora PID, należy dodać 1 w celu zmiany sposobu obliczania uchyby z SP-PV na PV-SP lub dodać 2 w celu zmiany polaryzacji sygnału ustawiającego CV z CV=sygnal wyjściowy regulatora PID na CV=-sygnal wyjściowy regulatora PID lub dodać 4 w celu zmiany sposobu różniczkowania z różniczkowania uchybu na różniczkowanie sygnału PV.</p>

Tabela 12-9. Zestawienie parametrów bloku funkcyjnego PID (kontynuacja)

	Opis																								
Manual Command - Sygnał sterujący w trybie ręcznym (13)	Wartość całkowita ze znakiem, równa sygnałowi ustawiającemu CV, jeżeli regulator PID pracuje w trybie automatycznym. Po przełączeniu regulatora w tryb ręcznego sterowania (Manual), wartość ta wykorzystywana jest do wyznaczenia wartości CV oraz wewnętrznej wartości bloku całkującego w obrębie przedziału wyznaczonego przez Górną i Dolną granicy wartości sygnału sterującego oraz Minimalny czas narastania sygnału sterującego.																								
Control Word - Słowo sterujące (14)	<p>Jest to parametr wewnętrzny, którego wartość jest zwykle równa 0.</p> <p>Jeżeli wartość logiczna bitu zerowego jest równa 1, blok funkcyjny PID sterowany jest przez to Słowo konfiguracyjne oraz przez inne wewnętrzne parametry SP, PV i CV (proszę porównać z opisem poniżej). Dzięki temu urządzenia zewnętrzne, takie jak na przykład komputer, mogą przejąć sterowanie od programu sterującego. Ostrzeżenie: Jeżeli sytuacja taka nie powinna wystąpić, należy upewnić się że wartość Słowa sterującego jest równa 0. Jeżeli wartość bitu zerowego jest równa 0, można na podstawie następujących 4 bitów odczytywać status styków wejściowych regulatora PID, tak długo jak długo doprowadzany jest sygnał wejściowy do tego bloku. Jest to słowo bitowe o strukturze dyskretnej (każdy z bitów posiada odrębne znaczenie), o następującym formacie:</p> <table border="1"> <thead> <tr> <th>Bit:</th> <th>Wartość słowa:</th> <th>Funkcja:</th> <th>Status lub działania zewnętrzne podejmowane po zmianie wartości bitu zerowego na 1:</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>1</td> <td>Bit 0</td> <td>Jeżeli jego wartość wynosi 0, monitorowane są wejścia stykowe bloku funkcyjnego.</td> </tr> <tr> <td>1</td> <td>2</td> <td>Manul</td> <td>Jeżeli 1, blok jest w trybie Manual, w przypadku wartości 0, w trybie Auto.</td> </tr> <tr> <td>2</td> <td>4</td> <td>Enable</td> <td>Standardowo wartość 1, w przeciwnym wypadku blok nie jest wywoływany.</td> </tr> <tr> <td>3</td> <td>8</td> <td>UP/Raise</td> <td>Jeżeli 1 i blok pracuje w trybie ręcznym, każde wykonanie bloku PID powoduje inkrementowanie sygnału CV.</td> </tr> <tr> <td>4</td> <td>16</td> <td>DN/Lower</td> <td>Jeżeli 1 i blok pracuje w trybie ręcznym, każde wykonanie bloku PID powoduje inkrementowanie sygnału CV.</td> </tr> </tbody> </table>	Bit:	Wartość słowa:	Funkcja:	Status lub działania zewnętrzne podejmowane po zmianie wartości bitu zerowego na 1:	0	1	Bit 0	Jeżeli jego wartość wynosi 0, monitorowane są wejścia stykowe bloku funkcyjnego.	1	2	Manul	Jeżeli 1, blok jest w trybie Manual , w przypadku wartości 0, w trybie Auto .	2	4	Enable	Standardowo wartość 1, w przeciwnym wypadku blok nie jest wywoływany.	3	8	UP/Raise	Jeżeli 1 i blok pracuje w trybie ręcznym, każde wykonanie bloku PID powoduje inkrementowanie sygnału CV.	4	16	DN/Lower	Jeżeli 1 i blok pracuje w trybie ręcznym, każde wykonanie bloku PID powoduje inkrementowanie sygnału CV.
Bit:	Wartość słowa:	Funkcja:	Status lub działania zewnętrzne podejmowane po zmianie wartości bitu zerowego na 1:																						
0	1	Bit 0	Jeżeli jego wartość wynosi 0, monitorowane są wejścia stykowe bloku funkcyjnego.																						
1	2	Manul	Jeżeli 1, blok jest w trybie Manual , w przypadku wartości 0, w trybie Auto .																						
2	4	Enable	Standardowo wartość 1, w przeciwnym wypadku blok nie jest wywoływany.																						
3	8	UP/Raise	Jeżeli 1 i blok pracuje w trybie ręcznym, każde wykonanie bloku PID powoduje inkrementowanie sygnału CV.																						
4	16	DN/Lower	Jeżeli 1 i blok pracuje w trybie ręcznym, każde wykonanie bloku PID powoduje inkrementowanie sygnału CV.																						
SP - SP (15)	Punkt pracy regulatora SP. (Nie konfigurowalny – wartość ustawiana i pamiętana przez sterownik). Jeżeli bit 0 ma wartość 1, wartość tego parametru musi być ustawiona z zewnątrz.																								
CV - CV (16)	Sygnał ustawiający CV. (Nie konfigurowalny - ustawiany i pamiętany przez sterownik).																								
PV - PV (17)	Wartość wielkości regulowanej PV. (Nie konfigurowalny - wartość ustawiana i pamiętana przez sterownik). Jeżeli bit 0 ma wartość 1, wartość tego parametru musi być ustawiona z zewnątrz.																								
Output - Polaryzacja sygnału ustawiającego (18)	(Nie konfigurowalny - ustawiany i pamiętany przez sterownik). Wartość całkowita ze znakiem reprezentująca bieżący stan parametru wyjściowego bloku funkcyjnego PID, przed zastosowaniem opcjonalnej konwersji sygnału. Jeżeli nie jest zaprogramowana inwersja sygnału ustawiającego i bit polaryzacji sygnału (pierwszy bit parametru konfiguracyjnego) ma wartość 0, wartość ta jest równa wartości sygnału ustawiającego CV. Jeżeli inwersja sygnału ustawiającego jest zaprogramowana i bit polaryzacji sygnału (pierwszy bit parametru konfiguracyjnego) ma wartość 1, wartość ta jest równa wartości sygnału ustawiającego CV, z przeciwnym znakiem.																								
Diff Term Storage - Dane robocze dla różniczkow. (19)	Rejestr wykorzystywany przez blok funkcyjny PID do pamiętania wartości pośrednich. <i>W rejestrze tym nie wolno zapisywać żadnych danych.</i>																								
Int Term Storage - Dane wewn. (20/21)	Rejestr wykorzystywany przez blok funkcyjny PID do pamiętania wartości pośrednich. <i>W rejestrze tym nie wolno zapisywać żadnych danych.</i>																								
Slew Term Storage - Dane wewn. (22)	Rejestr wykorzystywany przez blok funkcyjny PID do pamiętania wartości pośrednich. <i>W rejestrze tym nie wolno zapisywać żadnych danych.</i>																								
Clock - Zegar wewn. (23-25)	Czas, który upłynął od momentu ostatniego wykonania algorytmu PID. <i>W rejestrach tych nie wolno zapisywać żadnych danych.</i>																								
Rejestr do przechowywania wartości Y (26)	Parametr ten zawiera resztę z skalowania działki całkowania dla stabilnego, zerowego uchybu.																								
Lower and Upper Range - Górna i dolna gr. zakresu war par SP i PV (27/28)	Opcjonalne wartości całkowite w jednostkach bezwymiarowych sygnału ustawiającego PV, wykorzystywane do graficznej reprezentacji wartości SP i PV w oprogramowaniu LM90.																								
Zarezerwowane (29-34 i 35-39)	Rejestry 29-34 są zarezerwowane do użytku wewnętrznego, a rejestry 35-39 są zarezerwowane do użytku zewnętrznego. Są one zarezerwowane dla GE Fanuc i nie mogą być wykorzystywane w innym celu.																								

Parametry wewnętrzne w tablicy RefArray

Zgodnie z opisem przedstawionym w Tabeli 12-9 na poprzedniej stronie, blok PID odczytuje 13 parametrów użytkownika a pozostałe z 40 parametrów jest wykorzystywanych do wewnętrznych obliczeń. Standardowo nie ma potrzeby modyfikowania jakiegokolwiek z tych wartości. Jeżeli jednak blok PID wywoływany jest w trybie automatycznym po dłuższej przerwie, można zapisać w parametrze %Ref+ 23 za pomocą bloku funkcyjnego SVCREQ #16 bieżący czas zegara w celu uaktualnienia czasu ostatniego wykonania bloku PID, co pozwoli na uniknięcie skoku członu całkującego. Jeżeli wartość bitu 0 Słowa sterującego (%Ref + 14) została ustawiona na 1, następnie cztery bity Słowa sterującego będą wykorzystywane do sterowania stykami wejściowymi bloku PID (zgodnie z opisem zamieszczonym na poprzedniej stronie w Tabeli 12-9) oraz muszą być ustawione wartości Punktu pracy regulatora SP i wielkości regulowanej PV, ponieważ szczebel programu sterującego nie jest już w dalszym ciągu odpowiedzialny za sterowanie regulatorem PID.

Wybór algorytmu sterowania PID (PIDISA lub PIDIND) oraz wzmocnień

Blok PID można zaprogramować w oparciu o algorytm o niezależnych wyrazach (PID_INT) lub standardowy algorytm ISA (PID_ISA). Jedyna różnica polega na definicji współczynnika wzmocnienia dla bloku całkującego i różniczkującego. Aby zrozumieć tę różnicę należy mieć świadomość podanych poniżej zależności:

W obydwu algorytmach PID uchyb jest obliczany jako wartość wyrażenia SP-PV lub wyrażenia PV-SP jeżeli bit zerowy w Słowie konfiguracyjnym (%Ref+12) ma wartość równą 1. Zanegowana wartość uchybu może być stosowana jeżeli wartość sygnału ustawiającego CV ma zmieniać się w kierunku przeciwnym do wartości wielkości regulowanej PV (zwiększanie wartości CV przy zmniejszaniu się wartości PV), czyli jeśli wartości CV i PV nie mają się zmieniać w tym samym kierunku.

Uchyb = (SP - PV) lub (PV-SP) jeżeli bit zerowy Słowa konfiguracyjnego ma wartość 1.

Blok różniczkujący zależy standardowo od zmiany uchybu w czasie ostatniej realizacji algorytmu PID, blok może powodować dużą zmianę sygnału ustawiającego jeżeli wartość zadana uległa zmianie. Jeżeli taka właściwość nie jest pożądana, można ustawić trzeci bit Słowa konfiguracyjnego na 1, co spowoduje obliczanie wyrazu różniczkującego w oparciu o zmianę wartości zadanej PV. Wartość dt (przyrost czasu) jest wyznaczana poprzez odjęcie od bieżącego czasu sterownika czasu, który upłynął od momentu ostatniego wykonania algorytmu PID.

dt = Bieżący czas sterownika - czas, który upłynął od momentu ostatniego wykonania algorytmu PID

Blok różniczkujący = (Uchyb - Poprzedni uchyb)/dt lub (PV - poprzednia wartość PV)/dt jeżeli 3-i bit Słowa konfiguracyjnego ma wartość 1.

Algorytm PID o niezależnych wyrazach (PID_INT) oblicza sygnał ustawiający jako:

Sygnał ustawiający regulatora PID = $K_p * \text{Uchyb} + K_i * \text{Uchyb} * dt + K_d * \text{pochodna} + \text{CV Bias}$

Standardowy algorytm ISA (PID_ISA) korzysta z innego wzoru:

Sygnał ustawiający regulatora PID = $K_c * (\text{Uchyb} + \text{Uchyb} * dt/T_i + T_d * \text{pochodna}) + \text{CV Bias}$

gdzie K_c jest współczynnikiem wzmocnienia proporcjonalnego, T_i jest czasem całkowania a T_d jest czasem różniczkowania. Zaletą algorytmu ISA jest fakt, że zmiana parametru K_c powoduje zmianę wyrazu proporcjonalnego, jak również wyrazu różniczkującego i całkującego, co ułatwia dostrajanie zamkniętego układu regulacji. Jeżeli wzmocnienie regulatora PID wyrażone jest przy pomocy T_i i T_d , należy zastosować następujące wyrażenia

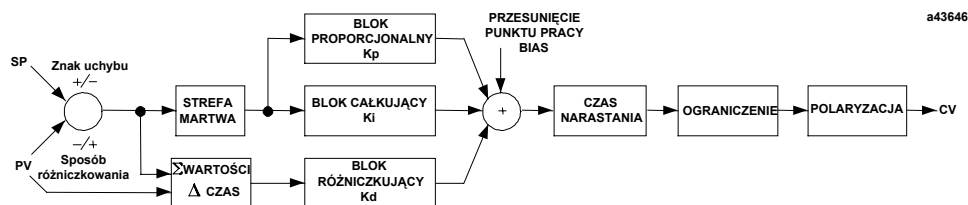
$K_p = K_c$ $K_i = K_c/T_i$ i $K_d = K_c/T_d$

do przekonwertowania ich do parametrów wejściowych użytkownika bloku PID.

Wymieniony powyżej składnik CV Bias jest dodatkowym składnikiem, niezależnym od pozostałych parametrów regulatora PID. Może on być potrzebny, jeżeli stosowany jest wyłącznie proporcjonalny współczynnik wzmocnienia K_p , a wartość sygnału ustawiającego CV ma być różna od zera, jeżeli wartość regulowana PV jest równa wartości zadanej SP i uchyb jest równy 0. Jeżeli wystąpi taka sytuacja, parametr CV zawiera żadaną wartość sygnału sterującego, jeżeli wartość regulowana PV jest równa wartości zadanej SP. Parametr CV Bias może być również wykorzystywany do sterowania ze sprzężeniem dodatnim, gdzie sygnał ustawiający CV danego regulatora PID jest zmieniany przez inny układ PID lub algorytm sterowania.

Jeżeli wykorzystywane jest wzmocnienie wyrazu całkującego K_i , parametr CV Bias jest zwykle równy 0, ponieważ blok całkujący automatycznie nie dopuszcza do otrzymania zerowej wartości sygnału ustawiającego. Wystarczy rozpocząć pracę w trybie ręcznym, a następnie za pomocą parametru Sygnał sterujący w trybie ręcznym (%Ref + 13) ustawić blok całkujący na żadaną wartość sygnału zadającego CV, po czym przejść do trybu sterowania automatycznego. Mechanizm ten pracuje również jeżeli K_i jest równe 0, za wyjątkiem faktu, że wartość wyrazu całkującego nie będzie zmieniana w zależności od wartości uchybu po przejściu do trybu automatycznego.

Zamieszczony poniżej schemat obrazuje pracę algorytmów PID:



Rysunek 12-4. Regulator PID o niezależnych wyrazach (PIDIND).

Algorytm ISA (PIDISA) jest bardzo podobny, różnica polega na wyeliminowaniu współczynnika K_p z współczynników K_i i K_d , przez co wzmocnienie wyrazu całkującego wynosi $K_p \cdot K_i$, a wzmocnienie wyrazu różniczkującego jest równe $K_p \cdot K_d$. Znak uchybu, sposób różniczkowania oraz polaryzacja są wybierane poprzez ustawienie odpowiednich wartości parametru użytkownika Słowo konfiguracyjne.

Wartości graniczne amplitudy i prędkości narastania sygnału zadającego CV

Blok PID nie wysyła obliczonego sygnału wyjściowego bezpośrednio do parametru CV. Obydwa algorytmy PID umożliwiają ograniczenie amplitudy i prędkości narastania sygnału ustawiającego PV. Maksymalna prędkość narastania jest wyznaczana poprzez podzielenie maksymalnej wartości wynoszącej 100% sygnału zadającego CV (32000) przez Minimalny czas narastania sygnału ustawiającego, o ile jego wartość jest większa od 0. Przykładowo, jeżeli Minimalny czas narastania sygnału sterującego wynosi 100 sekund, prędkość narastania jest równa 320 jednostek bezwymiarowych CV na sekundę. Jeżeli czas od momentu ostatniego wykonania wynosi 50 milisekund, nowa wartość sygnału ustawiającego CV nie może się zmienić o więcej niż $320 \cdot 50 / 1000$, czyli o 16 jednostek bezwymiarowych sygnału ustawiającego CV, względem poprzedniej wartości.

Sygnał wyjściowy CV jest następnie porównywany z Górną i Dolną granicą wartości sygnału zadającego. Jeżeli przekroczona zostanie jedna z tych wartości, przyjmowana jest graniczna wartość sygnału zadającego CV. Jeżeli nastąpiło przekroczenie zarówno wartości granicznych amplitudy jak i prędkości narastania sygnału, następuje dostosowanie wewnętrznej wartości wyrazu całkującego w celu uniknięcia przekraczania dopuszczalnych wartości.

Ostatecznie blok sprawdza polaryzację wyjściową (2 bit Słowa konfiguracyjnego o adresie %Ref+2) i zmienia znak sygnału, jeżeli wartość tego bitu jest równa 1.

CV= Sygnał wyjściowy regulatora PID z uwzględnieniem wartości granicznych lub –Sygnał wyjściowy regulatora PID z uwzględnieniem wartości granicznych, jeżeli ustawiono bit polaryzacji sygnału wyjściowego.

Jeżeli blok pracuje w trybie automatycznym, ostateczna wartość sygnału ustawiającego CV jest zapisywana do parametru Sygnał sterujący w trybie ręcznym (%Ref + 13). Jeżeli blok pracuje w trybie ręcznym, równanie zamkniętego układu regulacji PID jest pomijane, a wartość sygnału ustawiającego CV jest określana poprzez parametr Sygnał sterujący w trybie ręcznym, są jednak nadal sprawdzane wszystkie wartości graniczne amplitudy i prędkości narastania. Oznacza to, że przy pomocy parametru Sygnał sterujący w trybie ręcznym nie można określić wartości sygnału zadającego przekraczającej podane wartości graniczne, jak też nie jest możliwe narastanie sygnału z prędkością większą od określonej przy pomocy parametru Minimalny czas narastania sygnału.

Okres próbkowania bloku PID

Blok PID jest cyfrową implementacją analogowej funkcji sterowania, co powoduje, że czas próbkowania dt w równaniu zamkniętego układu regulacji nie jest nieskończenie mały, jak to ma miejsce w przypadku analogowych układów sterowania. Większość sterowanych procesów może być przybliżona jako wzmacnienie z inercją pierwszego lub drugiego rzędu, często z opóźnieniem czasowym. Blok PID wyznacza wartość zmiennej ustawiającej CV procesu oraz wykorzystuje bieżącą wartość wartości regulowanej PV do obliczania następnej wartości zmiennej CV. Kluczowym parametrem procesu jest stała czasowa, określająca prędkość narastania wielkości regulowanej PV w przypadku zmiany wielkości ustawiającej CV. Zgodnie z przedstawionym poniżej w punkcie "Metoda dostrajania współczynnika wzmacnienia regulatora PID" opisem, stała czasowa T_p+T_c dla systemu pierwszego rzędu jest opóźnieniem do momentu, kiedy wielkość regulowana PV dochodzi do 63% wartości końcowej, przy skokowej zmianie sygnału ustawiającego CV. Blok PID nie będzie w stanie sterować procesem, jeżeli Okres próbkowania jest większy od połowy globalnej stałej czasowej. Dłuższe okresy próbkowania nie zapewniają stabilności.

Okres próbkowania nie powinien być większy od globalnej stałej czasowej podzielonej przez 10 (lub w najgorszym przypadku przez 5). Przykładowo, jeżeli wydaje się że wartość regulowana PV dojdzie do około 2/3 swojej końcowej wartości w okresie 2 sekund, okres próbkowania nie powinien być mniejszy od 0.2 sekund, a w najgorszym przypadku 0.4 sekund. Z drugiej strony, okres próbkowania nie może być zbyt mały, jak na przykład nie może być mniejszy od globalnej stałej czasowej podzielonej przez 1000, bowiem składnik $K_i \cdot Uchyb \cdot dt$ regulatora PID będzie zaokrąglany w dół do zera. Przykładowo, w przypadku bardzo wolnego procesu, który zachodzi w ciągu 10 godzin, czyli 36000 sekund, w celu dojścia wartości regulowanej do poziomu 63% swojej wartości, okres próbkowania powinien wynosić 40 sekund lub więcej.

Jeżeli proces nie zachodzi bardzo szybko, zwykle nie jest konieczne wprowadzanie wartości 0 dla okresu próbkowania, w celu wykonywania algorytmu PID w każdym cyklu. Jeżeli kilka bloków PID ma ustawiony okres próbkowania o wartości większej od czasu trwania cyklu, czas ten może ulegać znacznym wahaniom, ponieważ w jednym czasie może być rozwiązywanych wiele algorytmów PID. Najprostszym rozwiązaniem jest zastosowanie tablicy bitowej z wędrującą jedyneką, sterującej dopływem sygnału do poszczególnych bloków PID.

Wyznaczanie charakterystyk procesu

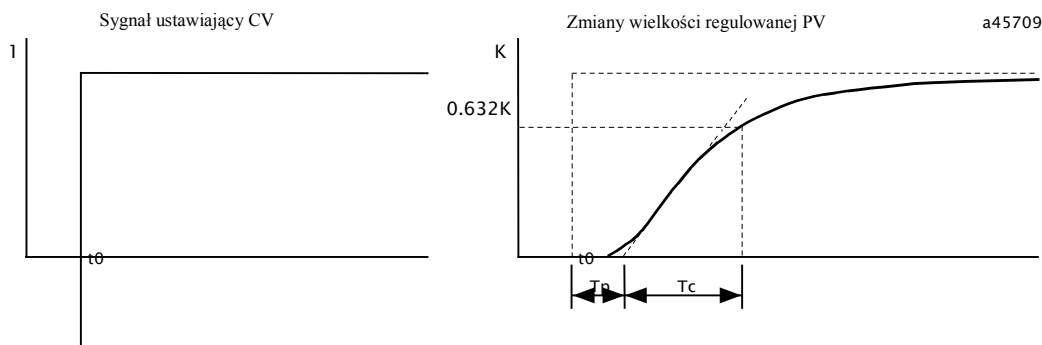
Współczynniki dla algorytmu PID, K_p , K_i i K_d są wyznaczone na podstawie charakterystyki sterowanego procesu. W czasie wyznaczania parametrów zamkniętego układu regulacji PID należy znaleźć odpowiedź na dwa podstawowe pytania:

1. Jak zmienia się wartość wartości regulowanej PV przy zmianie sygnału ustawiającego CV o stałą wartość lub jak duże jest wzmocnienie otwartego układu regulacji?
2. Jaka jest szybkość reakcji systemu lub jak szybko zmienia się wartość wielkości regulowanej PV po skokowej zmianie wielkości ustawiającej CV?

Wiele procesów może być przybliżonych przez współczynnik wzmocnienia procesu, stałą czasową pierwszego lub drugiego rzędu oraz czas opóźnienia. W dziedzinie częstotliwości, transmitancja dla systemu ze zwłoką czasową pierwszego rzędu ma postać:

$$PV(s)/CV(s) = G(s) = K \cdot e^{-T_p s} / (1 + T_c s)$$

Wykreślenie odpowiedzi skokowej w chwili t_0 w dziedzinie czasu daje krzywą zmiany w układzie otwartym:



Na podstawie krzywej zmiany wielkości regulowanej PV można wyznaczyć następujące parametry procesu:

K	Wzmocnienie w otwartym układzie sterowania procesem = końcowa zmiana wielkości regulowanej PV/ zmiana sygnału ustawiającego CV w chwili t_0 (zwrócić uwagę na brak indeksu przy współczynniku K).
T_p	Czas opóźnienia procesu lub czas po t_0 , po upływie którego wartość PV zacznie się zmieniać.
T_c	Stała czasowa procesu pierwszego rzędu, czas mierzony od T_p do momentu kiedy wartość regulowana PV uzyska 63.3% wartości ostatecznej.

Zwykle najszybszą metodą pomiaru tych parametrów jest wywołanie bloku PID w trybie ręcznym, z małymi zmianami sygnału ustawiającego CV, poprzez wprowadzanie odpowiednich wartości parametru Sygnał sterujący w trybie ręcznym i wykreślanie w funkcji czasu zmian wartości wielkości regulowanej PV. W przypadku wolnych procesów, czynność ta może być wykonywana ręcznie, natomiast w przypadku procesów szybszych, zachodzi potrzeba skorzystania z rejestratora danych lub komputera z odpowiednim oprogramowaniem. Wartość zmiany sygnału ustawiającego CV powinna być dostatecznie duża tak, aby powodowała dającą się zaobserwować zmianę wartości regulowanej PV, ale nie na tyle duża, aby doprowadzić do przerwania mierzonego procesu. Zmiana może przykładowo wynosić od 2 do 10% różnicy pomiędzy górną i dolną wartością graniczną sygnału ustawiającego.

Dostrajanie parametrów regulatora PID

Wszystkie parametry bloku PID zależą całkowicie od rodzaju sterowanego procesu, nie można zalecić żadnych wartości, które na pewno okażą się słuszne, można jednakże zastosować prostą, iteracyjną metodę w celu znalezienia akceptowalnego współczynnika wzmocnienia.

1. Ustawić wartość wszystkich parametrów na 0, a następnie ustawić Górną i dolną wartość sygnału sterującego na odpowiednio maksymalną i minimalną oczekiwaną wartość CV. Ustawić Okres próbkowania na wartość szacowanej globalnej stałej czasowej procesu podzielonej przez liczbę z przedziału (0-100)
2. Uruchomić blok w trybie ręcznym, a następnie wprowadzać do parametru Sygnał sterujący w trybie ręcznym (%Ref +13) różne wartości w celu sprawdzenia, czy wartość ustawiająca CV dochodzi do górnej i dolnej wartości granicznej. Zanotować wartości regulowanej PV dla pewnych wartości ustawiającej CV, a następnie przypisać je do wartości zadanej SP.
3. Ustawić małe wzmocnienie Kp, przykładowo $100 \cdot \frac{\text{Maksymalna wartość CV}}{\text{maksymalna wartość PV}}$, po czym przejść do trybu ręcznego. Zmienić skokowo wartość SP o 2 do 10% wartości maksymalnej wartości regulowanej PV i obserwować odpowiedzi wartości regulowanej PV. Zwiększyć wartość współczynnika wzmocnienia Kp jeżeli odpowiedź wartości regulowanej PV jest zbyt wolna lub też zmniejszyć ją, jeżeli zmiany wartości regulowanej PV są zbyt duże, albo oscylują, bez dochodzenia do stanu stabilności.
4. Po znalezieniu odpowiedniego współczynnika wzmocnienia Kp, zwiększać wartość Ki w celu znalezienia przeregulowania tłumionego do wartości ustalonej w ciągu 2 do 3 cykli. Może to wymagać zmniejszenia współczynnika Kp. Spróbować również zmienić skokową wielkość zmian oraz wprowadzić inne wartości sygnału ustawiającego CV.
5. Po znalezieniu odpowiednich współczynników Kp i Ki, spróbować zwiększać współczynnik Kd w celu szybszego uzyskiwania odpowiedzi na parametry wejściowe, jednak bez powstawania oscylacji. Współczynnik Kd jest bardzo często niepotrzebny oraz nie powinien być wykorzystywany w przypadku zakłóconego sygnału wielkości regulowanej PV.
6. Sprawdzić współczynnik wzmocnienia dla różnych wartości zadających SP oraz, jeżeli jest to potrzebne, określić strefę nieczułości i Minimalny czas narastania sygnału ustawiającego. W przypadku niektórych procesów konieczne może być ustawienie bitów Słowa konfiguracyjnego w celu zmiany znaku uchybu lub polaryzacji.

Dobór współczynników wzmocnienia za pomocą metody Zieglera i Nicholasa

Po wyznaczeniu trzech parametrów modelu procesu, K , T_p i T_c można wstępnie oszacować wartość współczynników wzmocnienia dla zamkniętego układu regulacji PID. Przedstawiona poniżej metoda, opracowana w 1942 roku przez Zieglera i Nicholasa, ma za zadanie zapewnienie dobrej reakcji systemu przy wzmocnieniach dających stosunek amplitud 1/4. Stosunek amplitud jest to stosunek drugiej wartości szczytowej do pierwszej wartości szczytowej odpowiedzi zamkniętego układu regulacji.

1. Obliczyć szybkość reakcji regulatora:

$$R = K/T_c$$

2. Gdy regulator pracuje jedynie jako regulator proporcjonalny, policzyć współczynnik wzmocnienia wyrazu proporcjonalnego K_p :

$$K_p = 1/(R * T_p) = T_c/(K * T_p)$$

3. Gdy regulator pracuje jako proporcjonalny i całkujący, skorzystać z równania :

$$K_p = 0.9/(R * T_p) = 0.9 * T_c/(K * T_p)$$

$$K_i = 0.3 * K_p/T_p$$

4. Gdy regulator pracuje jako proporcjonalny, całkujący i różniczkujący, zastosować równanie :

$$K_p = G/(R * T_p)$$

gdzie G ma wartość z zakresu od 1.2 do 2.0

$$K_i = 0.5 * K_p/T_p$$

$$K_d = 0.5 * K_p * T_p$$

5. Sprawdzić, czy okres próbkowania ma wartość z zakresu $(T_p + T_c)/10$ do $(T_p + T_c)/1000$

Inna metoda "Idealnego dostrojenia" ma za zadanie zapewnić jak najlepszej odpowiedzi na zmiany wartości zadanej SP, opóźnionej wyłącznie przez opóźnienie procesu T_p lub czas martwy:

$$K_p = 2 * T_c/(3 * K * T_p)$$

$$K_i = T_c$$

$$K_d = K_i/4$$

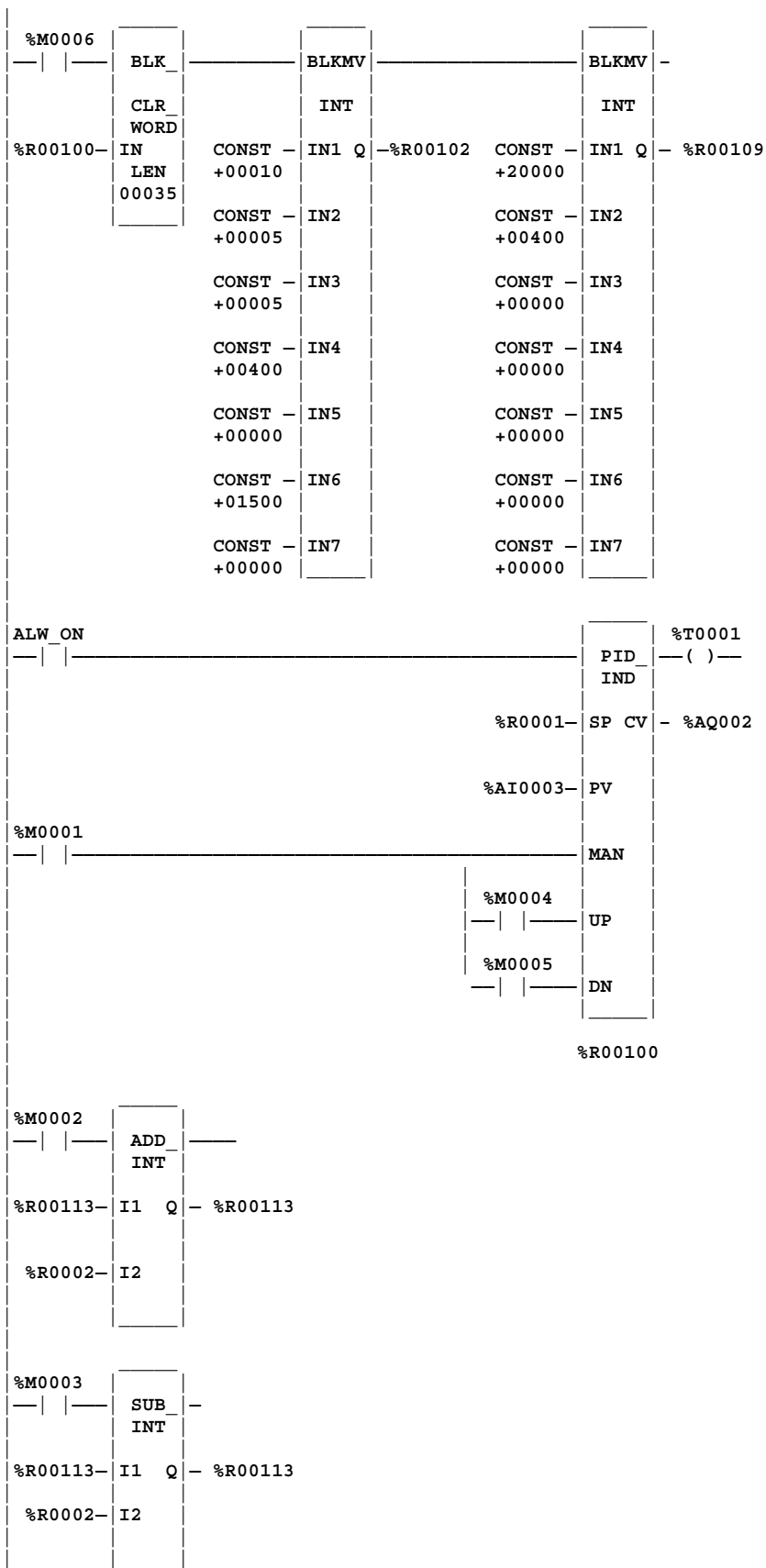
jeżeli wykorzystywany jest składnik różniczkujący

Po wyznaczeniu początkowych wartości współczynników wzmocnienia, można je przekonwertować na parametry użytkownika, będące liczbami całkowitymi. W celu uniknięcia problemów ze skalowaniem, wzmocnienie procesu K powinno być obliczane jako zmiana w jednostkach bezwymiarowych wartości regulowanej PV podzielonej przez skokową zmianę wartości wyjściowej CV w jednostkach bezwymiarowych CV, a nie w jednostkach inżynierskich parametrów PV czy CV. Również czas powinien być podawany w sekundach. Po wyznaczeniu współczynników K_p , K_i i K_d , współczynniki K_p i K_d można pomnożyć przez 100 i wprowadzić je jako wartości całkowite, a współczynnik K_i należy pomnożyć przed wprowadzeniem do tabeli parametrów użytkownika %RefArray przez 1000.

Przykład prostego wywołania regulatora PID

W zamieszczonym poniżej przykładzie, Okres próbkowania wynosi 100 milisekund, współczynnik wzmocnienia K_p jest równy 4.00 a współczynnik wzmocnienia K_i jest równy 1.500. Wartość zadana SP zapamiętana w %R1, sygnał ustawiający CV jest zapisywany w %AQ2, a wielkość regulowana zapisywana jest do %AI3. Muszą być również określone: górna i dolna granica sygnału ustawiającego, w niniejszym przykładzie są to 20000 i 400 oraz opcjonalnie wprowadzono górną i dolną granice strefy nieczułości +5 i -5. Tablica RefArray zawierająca 40 słów rozpoczyna się od adresu %R100. Zwykle parametry użytkownika są wprowadzane do tablicy RefArray za pomocą klawisza **F10**, ale można również ponownie zainicjować wartości 14 parametrów, począwszy od adresu %R102 (%Ref+2) w oparciu o stałe, pamiętane w programie sterującym.

Blok funkcyjny PID można przełączyć w tryb ręcznego sterowania za pomocą zmiennej %M1, co pozwala na zmianę parametru Sygnał sterujący w trybie ręcznym (rejestr %Ref+13, czyli w niniejszym przykładzie %R0113). Bity %M4 lub %M5 umożliwiają zwiększenie lub zmniejszenie wartości zmiennej %R113 o 1 przy każdym rozwiązywaniu algorytmu PID, w okresie co 100 milisekund. W celu przyśpieszenia pracy w trybie ręcznym, za pomocą bitów %M2 i %M3 można w każdym cyklu pracy sterownika dodawać i odejmować wartość zapisaną w %R2 do wartości %R113. Sygnał wyjściowy %T1 jest wysyłany pod warunkiem poprawnego wykonania algorytmu regulatora PID.



Sterowniki serii 90-30, 90-20 i Micro udostępniają szereg funkcji i bloków funkcyjnych. W Załączniku A podano rozmiar pamięci w bajtach oraz czas wykonywania w mikrosekundach każdej z instrukcji. Rozmiar pamięci to liczba bajtów, zajmowanych przez każdą z instrukcji w drabinie logicznej programu sterującego.

Dla każdego elementu logicznego w tabelicy podano dwa rodzaje czasów wykonywania:

Czas wykonywania	Opis
W stanie aktywnym	Czas wymagany na realizację elementu logicznego w sytuacji, gdy doprowadzany jest do niego sygnał wejściowy i element ten przesyła sygnał wyjściowy. Zwykle, najkrótsze czasy są uzyskiwane jeżeli blok umieszczony jest w pamięci RAM programu sterującego (zorientowanej rejestrowo), a nie w pamięci dyskretnej).
W stanie nieaktywnym	Czas wymagany na realizację elementu logicznego w sytuacji, gdy doprowadzany jest do niego sygnał wejściowy, lecz element nie jest aktywny, jak na przykład przełącznik czasowy przy dopływie sygnału zerującego.

Uwaga

Liczniki i przełączniki czasowe uaktualnianie są za każdym razem ich wywołania w programie sterującym, wartość licznika jest zwiększana o czas trwania ostatniego cyklu,
a wartość przełączników czasowych o jeden.

Uwaga

W przypadku jednostek centralnych 350, 351, 352 i 360, czasy wykonywania są identyczne, za wyjątkiem funkcji MOVE, której czas wykonywania przez jednostkę centralną CPU350 jest inny - proszę porównać z uwagą podaną pod tabelą, na stronie A-6.

Tabela A-1. Czasy wykonywania instrukcji, modele standardowe

Grupa funkcji	Funkcja	W stanie aktywnym				W stanie nieaktywnym				Przyrost czasu				Rozm
		311	313	331	340/41	311	313	331	340/41	311	313	331	340/41	
Przełączniki czasowe	Przełącznik czasowy załączający z pamięcią	146	81	80	42	105	39	38	21	-	-	-	-	15
	Przełącznik czasowy wyłączający	98	47	44	23	116	63	58	32	-	-	-	-	9
	Przełącznik czasowy załączający	122	76	75	40	103	54	53	30	-	-	-	-	15
Liczniki	Licznik dodający	137	70	69	36	130	63	62	33	-	-	-	-	11
	Licznik odejmujący	136	70	69	37	127	61	61	31	-	-	-	-	11
Funkcje matematycz.	Dodawanie (INT)	76	47	46	24	41	0	1	0	-	-	-	-	13
	Dodawanie (DINT)	90	60	60	34	41	1	0	0	-	-	-	-	13
	Odejmowanie (INT)	75	46	45	25	41	0	1	0	-	-	-	-	13
	Odejmowanie (DINT)	92	62	62	34	41	1	0	0	-	-	-	-	13
	Mnożenie (INT)	79	49	50	28	41	0	1	0	-	-	-	-	13
	Mnożenie (DINT)	108	80	101	43	41	1	0	0	-	-	-	-	13
	Dzielenie (INT)	79	51	50	27	41	0	1	0	-	-	-	-	13
	Dzielenie (DINT)	375	346	348	175	41	1	0	0	-	-	-	-	13
	Dzielenie Modulo (INT)	78	51	49	27	41	0	1	0	-	-	-	-	13
	Dzielenie Modulo (DINT)	134	103	107	54	41	1	0	0	-	-	-	-	13
	Pierwiastek kwadratowy (INT)	153	124	123	65	42	0	1	0	-	-	-	-	9
	Pierwiastek kwadratowy (DINT)	268	239	241	120	42	0	0	1	-	-	-	-	9
	Relacje	Równy (INT)	66	35	36	19	41	1	1	0	-	-	-	-
Równy (DINT)		86	56	54	29	41	1	0	0	-	-	-	-	9
Różny (INT)		67	39	35	22	41	1	1	0	-	-	-	-	9
Różny (DINT)		81	51	51	28	41	1	0	0	-	-	-	-	9
Większy (INT)		64	33	35	20	41	1	1	0	-	-	-	-	9
Większy (DINT)		89	59	58	32	41	1	0	0	-	-	-	-	9
Większy lub równy (INT)		64	36	34	19	41	1	1	0	-	-	-	-	9
Większy lub równy (DINT)		87	58	57	30	41	1	0	0	-	-	-	-	9
Mniejszy (INT)		66	35		19	41	1	1	0	-	-	-	-	9
Mniejszy (DINT)		87	57		30	41	1	1	0	-	-	-	-	9
Mniejszy lub równy (INT)		66	36	34	21	41	1	1	0	-	-	-	-	9
Mniejszy lub równy (DINT)		86	57	56	31	41	1	1	0	-	-	-	-	9
Zakres (INT)		92	58	54	29	46	1	0	1	-	-	-	-	15
Zakres (DINT)		106	75	57	37	45	0	0	0	-	-	-	-	15
Zakres (WORD)		93	60	54	29	0	0	0	0	-	-	-	-	15

- Uwagi:**
1. Czasy (w mikrosekundach) podano dla jednostek centralnych 311, 313, 340 i 341 (331 z oprogramowaniem systemowym ver. 7), dla oprogramowania Logicmaster 90-30/20 w wersji 5.01.
 2. W przypadku funkcji wykonujących operacje na tablicach przyrost czasu podany jest w jednostkach o zadanej długości: dla funkcji do operacji bitowych w mikrosekundach/bit, a dla funkcji do przeprowadzania operacji na danych w mikrosekundach/ liczbę bitów lub słów.
 3. Czas w stanie aktywnym podano dla modułów o pojedynczej długości typu %R, %AI i %AQ.
 4. Czas dla COMMREQ został zmierzony pomiędzy jednostką centralną a modulem HSC.
 5. W przypadku funkcji DOIO jest to czas wysyłania parametrów do modułu z wyjściami dyskretnymi.
 6. Jeżeli istnieje więcej niż jedna możliwość, podany czas jest czasem najmniej korzystnym.

Tabela A-1. Czasy wykonywania instrukcji, modele standardowe - kontynuacja

Grupa funkcji	Funkcja	W stanie aktywnym				W stanie nieaktywnym				Przyrost czasu				Rozm.
		311	313	331	340/41	311	313	331	340/41	311	313	331	340/41	
Operacje na ciągach bitów	Logiczne AND	67	37	37	22	42	0	0	1	-	-	-	-	13
	Logiczne OR	68	38	38	21	42	0	0	1	-	-	-	-	13
	Alternatywa wyłączająca OR	66	38	37	20	42	0	1	1	-	-	-	-	13
	Negacja logiczna NOT	62	32	31	17	42	0	1	1	-	-	-	-	9
	Przesunięcie słowa bitowego w lewo	139	89	90	47	74	26	23	13	11.61	11.61	12.04	6.29	15
	Przesunięcie słowa bitowego w prawo	135	87	85	45	75	26	24	13	11.63	11.62	12.02	6.33	15
	Przesunięcie słowa bitowego w lewo w obiegu zamkniętym	156	127	126	65	42	1	1	0	11.70	11.78	12.17	6.33	15
	Przesunięcie słowa bitowego w prawo w obiegu zamkniętym	146	116	116	62	42	1	1	0	11.74	11.74	12.13	6.27	15
	Lokalizowanie pierwszego bitu o wartości 1	102	72	49	38	42	1	0	0	-	-	-	-	13
	Ustawianie wartości pojedynczego bitu na 0	68	38	35	21	42	1	1	1	-	-	-	-	13
	Sprawdzanie wartości pojedynczego bitu	79	49	51	28	41	0	0	1	-	-	-	-	13
	Ustawianie wartości pojedynczego bitu na 1	67	37	37	20	42	0	0	0	-	-	-	-	13
	Porównanie z maskowaniem (WORD)	217	154	141	74	107	44	39	21	-	-	-	-	25
	Porównanie z maskowaniem (DWORD)	232	169	156	83	108	44	39	22	-	-	-	-	25
	Przesyłanie danych	Przemieszczanie (INT)	68	37	39	20	43	0	0	0	1.62	1.62	5.25	1.31
Przemieszczanie (BIT)		94	62	64	35	42	0	0	0	12.61	12.64	12.59	6.33	13
Przemieszczanie (WORD)		67	37	40	20	41	0	0	0	1.62	1.63	5.25	1.31	13
Przemieszczanie (INT)		76	48	50	28	59	30	30	16	-	-	-	-	27
Przemieszczanie (WORD)		76	48	49	29	59	29	28	15	-	-	-	-	27
Zerowanie bloku pamięci		56	28	27	14	43	0	0	0	1.35	1.29	1.40	0.78	9
Rejestr przemieszczający (BIT)		201	153	153	79	85	36	34	18	0.69	0.68	0.71	0.37	15
Rejestr przemieszczający (WORD)		103	53	52	29	73	25	23	12	1.62	1.62	2.03	1.31	15
Przemieszczanie jedyńki		165	101	99	53	96	31	29	16	0.07	0.07	0.08	0.05	15
COMMREQ	1317	1272	1489	884	41	2	0	0	-	-	-	-	13	
Operacje na tablicach	Kopiowanie danych													
	INT	230	201	177	104	72	41	40	20	1.29	1.15	10.56	2.06	21
	DINT	231	202	181	105	74	44	42	23	3.24	3.24	10.53	2.61	21
	BIT	290	261	229	135	74	43	42	23	-0.03	-0.03	-0.01	0.79	21
	BYTE	228	198	176	104	74	42	42	23	0.81	0.82	8.51	1.25	21
	WORD	230	201	177	104	72	41	40	20	1.29	1.15	10.56	2.06	21
	Szukanie wartości zadanej													
	INT	197	158	123	82	78	39	37	20	1.93	1.97	2.55	1.55	19
	DINT	206	166	135	87	79	38	36	21	4.33	4.34	4.55	2.44	19
	BYTE	179	141	117	74	78	38	36	21	1.53	1.49	1.83	1.03	19
	WORD	197	158	123	82	78	39	37	20	1.93	1.97	2.55	1.55	19

- Uwagi:**
1. Czasy (w mikrosekundach) podano dla jednostek centralnych 311, 313, 340 i 341 (331 z oprogramowaniem systemowym wer. 7), dla oprogramowania Logicmaster 90-30/20 w wersji 5.01.
 2. W przypadku funkcji wykonujących operacje na tablicach przyrost czasu podany jest w jednostkach o zadanej długości: dla funkcji do operacji bitowych w mikrosekundach/bit, a dla funkcji do przeprowadzania operacji na danych w mikrosekundach/ liczbę bitów lub słów.
 3. Czas w stanie aktywnym podano dla modułów o pojedynczej długości typu %R, %AI i %AQ.
 4. Czas dla COMMREQ został zmierzony pomiędzy jednostką centralną a modulem HSC.
 5. W przypadku funkcji DOIO jest to czas wysyłania parametrów do modułu z wyjściami dyskretnymi.
 6. Jeżeli istnieje więcej niż jedna możliwość, podany czas jest czasem najmniej korzystnym.
 7. W przypadku instrukcji dla których w kolumnie Przyrost czasu podano wartość, wartość tę należy pomnożyć przez (Długość - 1), a następnie dodać ten wynik do czasu podstawowego.

Tabela A-1. Czasy wykonywania instrukcji, modele standardowe - kontynuacja

Grupa funkcji	Element	W stanie aktywnym				W stanie nieaktywnym				Przyrost czasu				Rozm.
		311	313	331	340/41	311	313	331	340/41	311	313	331	340/41	
	Szukanie wartości różnej													
	INT	198	159	124	83	79	39	36	21	1.93	1.93	2.48	1.52	19
	DINT	201	163	132	84	79	37	35	21	6.49	6.47	6.88	3.82	19
	BYTE	179	141	117	73	79	38	36	19	1.54	1.51	1.85	1.05	19
	WORD	198	159	124	83	79	39	36	21	1.93	1.93	2.48	1.52	19
	Szukanie wartości większej													
	INT	198	160	125	82	79	37	38	19	3.83	3.83	4.41	2.59	19
	DINT	206	167	135	88	78	38	36	20	8.61	8.61	9.03	4.88	19
	BYTE	181	143	118	73	79	37	36	19	3.44	3.44	3.75	2.03	19
	WORD	198	160	125	82	79	37	38	19	3.83	3.83	4.41	2.59	19
	Szukanie wartości większej lub równej													
	INT	197	160	124	83	77	38	36	20	3.86	3.83	4.45	2.52	19
	DINT	205	167	136	87	80	39	36	21	8.62	8.61	9.02	4.87	19
	BYTE	180	142	118	75	79	37	37	20	3.47	3.44	3.73	2.00	19
	WORD	197	160	124	83	77	38	36	20	3.86	3.83	4.45	2.52	19
	Szukanie wartości mniejszej													
	INT	199	159	124	84	78	38	36	20	3.83	3.86	4.48	2.48	19
	DINT	206	168	135	87	79	38	38	19	8.62	8.60	-1.36	4.88	19
	BYTE	181	143	119	75	80	38	37	20	3.44	3.44	3.75	2.00	19
	WORD	199	159	124	84	78	38	36	20	3.83	3.86	4.45	2.48	19
	Szukanie wartości mniejszej lub równej													
	INT	200	158	124	82	79	38	37	21	3.79	3.90	4.45	2.55	19
	DINT	207	167	137	88	78	39	37	19	8.60	8.61	9.01	4.86	19
	BYTE	180	143	119	74	78	40	37	19	3.46	3.44	3.73	2.02	19
WORD	200	158	124	82	79	38	37	21	3.79	3.90	4.45	2.55	19	
Funkcje konwersji	Konwersja na dane typu BCD-4	74	46	39	25	42	1	1	1	-	-	-	-	9
	Konwersja danych typu BCD na dane typu INT	77	50	34	25	42	1	1	1	-	-	-	-	9

- Uwagi:**
1. Czasy (w mikrosekundach) podano dla jednostek centralnych 311, 313, 340 i 341 (331 z oprogramowaniem systemowym ver. 7), dla oprogramowania Logicmaster 90-30/20 w wersji 5.01.
 2. W przypadku funkcji wykonujących operacje na tablicach przyrost czasu podany jest w jednostkach o zadanej długości: dla funkcji do operacji bitowych w mikrosekundach/bit, a dla funkcji do przeprowadzania operacji na danych w mikrosekundach/ liczbę bitów lub słów.
 3. Czas w stanie aktywnym podano dla modułów o pojedynczej długości typu %R, %AI i %AQ.
 4. Czas dla COMMREQ został zmierzony pomiędzy jednostką centralną a modulem HSC.
 5. W przypadku funkcji DOIO jest to czas wysyłania parametrów do modułu z wyjściami dyskretnymi.
 6. Jeżeli istnieje więcej niż jedna możliwość, podany czas jest czasem najmniej korzystnym.
 7. W przypadku instrukcji dla których w kolumnie Przyrost czasu podano wartość, wartość tę należy pomnożyć przez (Długość - 1), a następnie dodać ten wynik do czasu podstawowego.

Tabela A-1. Czasy wykonywania instrukcji, modele standardowe - kontynuacja

Grupa funkcji	Funkcja	W stanie aktywnym				W stanie nieaktywnym				Przyrost czasu				Rozm.
		311	313	331	340/41	311	313	331	340/41	311	313	331	340/41	
Funkcje sterujące	Wywołanie podprogramu	155	93	192	85	41	0	0	0	-	-	-	-	7
	Natychmiastowe uaktualnienie stanu wejść/wyjść	309	278	323	177	38	1	0	0	-	-	-	-	12
	Algorytm PID-ISA	1870	1827	1812	929	91	56	82	30	-	-	-	-	15
	Algorytm PID-IND	2047	2007	2002	1017	91	56	82	30	-	-	-	-	15
	Instrukcje END	-	-	-	-	-	-	-	-	-	-	-	-	-
	SVCREQ:													
	# 6	93	54	63	45	41	2	0	0	-	-	-	-	9
	# 7 (Odczyt)	-	37	309	161	-	2	0	0	-	-	-	-	9
	# 7 (Ustawianie)	-	37	309	161	-	2	0	0	-	-	-	-	9
	#14	447	418	483	244	41	2	0	0	-	-	-	-	9
	#15	281	243	165	139	41	2	0	0	-	-	-	-	9
	#16	131	104	115	69	41	2	0	0	-	-	-	-	9
	#18	-	56	300	180	-	2	0	0	-	-	-	-	9
	#23	1689	1663	1591	939	43	1	0	0	-	-	-	-	9
	#26//30*	1268	1354	6680	3538	42	0	0	0	-	-	-	-	9
	#29	-	-	55	41	-	-	1	0	-	-	-	-	9
	Para zagnieżdżonych instrukcji MCR/ENDMCR	135	73	68	39	75	25	21	12	-	-	-	-	8

- Uwagi:**
1. Czasy (w mikrosekundach) podano dla jednostek centralnych 311, 313, 340 i 341 (331 z oprogramowaniem systemowym wer. 7), dla oprogramowania Logicmaster 90-30/20 w wersji 5.01.
 2. W przypadku funkcji wykonujących operacje na tablicach przyrost czasu podany jest w jednostkach o zadanej długości: dla funkcji do operacji bitowych w mikrosekundach/bit, a dla funkcji do przeprowadzania operacji na danych w mikrosekundach/ liczbę bitów lub słów.
 3. Czas w stanie aktywnym podano dla modułów o pojedynczej długości typu %R, %AI i %AQ.
 4. Czas dla COMMREQ został zmierzony pomiędzy jednostką centralną a modulem HSC.
 5. W przypadku funkcji DOIO jest to czas wysyłania parametrów do modułu z wyjściami dyskretnymi.
 6. Jeżeli istnieje więcej niż jedna możliwość, podany czas jest czasem najmniej korzystnym.
 7. W przypadku instrukcji dla których w kolumnie Przyrost czasu podano wartość, wartość tę należy pomnożyć przez (Długość - 1), a następnie dodać ten wynik do czasu podstawowego.

Tabela A-2. Czasy wykonywania instrukcji, modele zaawansowane

Grupa funkcji	Funkcja	W stanie aktywnym	W stanie nieaktywnym	Przyrost czasu	W stanie aktywnym	W stanie nieakt.	Przyrost czasu	Rozm.	
		350/351/36x	350/351/36x	350/351/36x	352	352	352		
Przełączniki czasowe	Przełącznik czasowy z pamięcią	4	6	–	4	5	–	15	
	Przełącznik	3	3	–	2	2	–	15	
	Przełącznik czasowy bez pamięci, z zanegowanym wejściem	3	3	–	3	2	–	15	
Liczniki	Licznik dodający	1	3	–	2	2	–	13	
	Licznik odejmujący	3	3	–	1	2	–	13	
Funkcje matematyczne	Dodawanie (INT)	2	0	–	1	0	–	13	
	Dodawanie (DINT)	2	0	–	2	0	–	19	
	Dodawanie (REAL)	52	0	–	33	0	–	17	
	Odejmowanie (INT)	2	0	–	1	0	–	13	
	Odejmowanie (DINT)	2	0	–	2	0	–	19	
	Odejmowanie (REAL)	53	0	–	34	0	–	17	
	Mnożenie (INT)	21	0	–	21	0	–	13	
	Mnożenie (DINT)	24	0	–	24	0	–	19	
	Mnożenie (REAL)	68	1	–	38	1	–	17	
	Dzielenie (INT)	22	0	–	22	0	–	13	
	Dzielenie (DINT)	25	0	–	25	0	–	19	
	Dzielenie (REAL)	82	2	–	36	2	–	17	
	Dzielenie Modulo (INT)	21	0	–	21	0	–	13	
	Dzielenie Modulo (DINT)	25	0	–	25	0	–	19	
	Pierwiastek kwadratowy (INT)	42	1	–	41	1	–	10	
	Pierwiastek kwadratowy (DINT)	70	0	–	70	0	–	13	
	Pierwiastek kwadratowy (REAL)	137	0	–	35	0	–	11	
	Funkcje trygonom.	SIN (REAL)	360	0	–	32	0	–	11
		COS (REAL)	319	0	–	29	0	–	11
TAN (REAL)		510	1	–	32	1	–	11	
ASIN (REAL)		440	0	–	45	0	–	11	
ACOS (REAL)		683	0	–	63	0	–	11	
ATAN (REAL)		264	1	–	33	1	–	11	
Funkcje logarytmiczne	LOG (REAL)	469	0	–	32	0	–	11	
	LN (REAL)	437	0	–	32	0	–	11	
Funkcje wykładnicze	EXP	639	0	–	42	0	–	11	
	EXPT	89	1	–	54	1	–	17	
Konwersja wartości kąta	Konwersja radianów na stopnie	65	1	–	32	1	–	11	
	Konwersja stopni na radiany	59	0	–	32	0	–	11	

- Uwagi:**
1. Czasy (w mikrosekundach) podano dla jednostek centralnych 311, 313, 340 i 341 (331 z oprogramowaniem systemowym ver. 7), dla oprogramowania Logicmaster 90-30/20 w wersji 5.01.
 2. W przypadku funkcji wykonujących operacje na tablicach przyrost czasu podany jest w jednostkach o zadanej długości: dla funkcji do operacji bitowych w mikrosekundach/ bit, a dla funkcji do przeprowadzania operacji na danych w mikrosekundach/ liczbę bitów lub słów.
 3. Czas w stanie aktywnym podano dla modułów o pojedynczej długości typu %R, %AI i %AQ.
 4. Czas dla COMMREQ został zmierzony pomiędzy jednostką centralną a modułem HSC.
 5. W przypadku funkcji DOIO jest to czas wysyłania parametrów do modułu z wyjściami dyskretnymi.
 6. Jeżeli istnieje więcej niż jedna możliwość, podany czas jest czasem najmniej korzystnym.

Tabela A-2. Czasy wykonywania, modele zaawansowane - kontynuacja

Grupa funkcji	Funkcja	W stanie aktywnym	W stanie nieaktywn.	Przyrost czasu	W stanie aktywnym	W stanie nieaktywn.	Przyrost czasu	Rozm.
		350/351/36x	350/351/36x	350/351/36x	352	352	352	
Relacje	Równy (INT)	1	0	–	1	0	–	10
	Równy (DINT)	2	0	–	2	0	–	16
	Równy (REAL)	57	0	–	28	0	–	14
	Różny (INT)	1	0	–	1	0	–	10
	Różny (DINT)	1	0	–	1	0	–	16
	Różny (REAL)	62	0	–	31	0	–	14
	Większy (INT)	1	0	–	1	0	–	10
	Większy (DINT)	1	0	–	1	0	–	16
	Większy (REAL)	57	0	–	32	0	–	14
	Większy lub równy (INT)	1	0	–	1	0	–	10
	Większy lub równy (DINT)	1	0	–	1	0	–	10
	Większy lub równy (REAL)	57	1	–	31	1	–	14
	Mniejszy (INT)	1	0	–	1	0	–	10
	Mniejszy (DINT)	1	0	–	1	0	–	16
	Mniejszy (REAL)	58	1	–	36	1	–	14
	Mniejszy lub równy (INT)	1	0	–	1	0	–	10
	Mniejszy lub równy (DINT)	3	0	–	3	0	–	16
	Mniejszy lub równy (REAL)	37	0	–	37	0	–	14
	Zakres (INT)	2	1	–	2	1	–	13
	Zakres (DINT)	2	1	–	2	1	–	22
Zakres (WORD)	1	0	–	1	0	–	13	
Operacje na bitach	Logiczne AND	2	0	–	2	0	–	13
	Logiczne OR	2	0	–	2	0	–	13
	Alternatywa wyłączająca OR	1	0	–	1	0	–	13
	Negacja logiczna NOT	1	0	–	1	0	–	10
	Przesunięcie słowa bitowego w lewo	31	1	1.37	31	1	1.37	16
	Przesunięcie słowa bitowego w prawo	28	0	3.03	28	0	3.03	16
	Przesunięcie słowa bitowego w lewo w obiegu zamkniętym	25	0	3.12	25	0	3.12	16
	Przesunięcie słowa bitowego w prawo w obiegu zamkniętym	25	0	4.14	25	0	4.14	16
	Lokalizowanie pierwszego bitu o wartości 1	20	1	–	20	1	–	13
	Ustawianie wartości pojedynczego bitu na 0	20	0	–	20	0	–	13
	Sprawdzanie wartości pojedynczego bitu	20	0	–	20	0	–	13
	Ustawianie wartości pojedynczego bitu na 1	19	1	–	19	1	–	13
	Porównanie z maskowaniem (WORD)	52	0	–	52	0	–	25
	Porównanie z maskowaniem (DWORD)	50	0	–	49	0	–	25

Uwagi:

1. Czasy (w mikrosekundach) podano dla jednostek centralnych 351 i 352, dla oprogramowania Logicmaster 90-30/20 w wersji 7.
2. W przypadku funkcji wykonujących operacje na tablicach przyrost czasu podany jest w jednostkach o zadanej długości: dla funkcji do operacji bitowych w mikrosekundach/bit, a dla funkcji do przeprowadzania operacji na danych w mikrosekundach/ liczbę bitów lub słów.
3. Czas w stanie aktywnym podano dla modułów o pojedynczej długości typu %R, %AI i %AQ.
4. Czas dla COMMREQ został zmierzony pomiędzy jednostką centralną a modulem HSC.
5. W przypadku funkcji DOIO jest to czas wysyłania parametrów do modułu z wyjściami dyskretnymi.
6. Jeżeli istnieje więcej niż jedna możliwość, podany czas jest czasem najmniej korzystnym.
7. W przypadku instrukcji dla których w kolumnie Przyrost czasu podano wartość, wartość tę należy pomnożyć przez (Długość - 1), a następnie dodać ten wynik do czasu podstawowego.

Tabela A-2. Czasy wykonywania, modele zaawansowane - kontynuacja

Grupa funkcji	Funkcja	W stanie aktywnym	W stanie nieaktywnym	Przyrost czasu	W stanie aktywnym	W stanie nieaktyw.	Przyrost czasu	Rozm.
		350/351/36X	350/351/36X	350/351/36X	352	352	352	
Przesyłanie danych	Przemieszczanie (INT)	2	0	0.41	2	0	0.41	10
	Przemieszczanie (BIT)	28	0	4.98	28	0	4.98	13
	Przemieszczanie (WORD)	2	0	0.41	2	0	0.41	10
	Przemieszczanie (REAL)	24	1	0.82	24	1	0.82	13
	Przemieszczanie bloku (INT)	2	0	–	2	0	–	28
	Przemieszczanie bloku (WORD)	4	4	–	3	0	–	28
	Przemieszczanie bloku (REAL)	41	0	–	41	0	–	13
	Zerowanie bloku pamięci	1	0	0.24	1	0	0.24	11
	Rejestr przemieszczający (BIT)	49	0	0.23	46	0	0.23	16
	Rejestr przemieszczający (WORD)	27	0	0.41	27	0	0.41	16
	Przemieszczanie jedynki	38	22	0.02	38	22	0.02	16
	COMMREQ	765	0	–	765	0	–	13
Operacje na bitach	Kopiowanie danych							
	INT	54	0	0.97	54	0	0.97	22
	DINT	54	0	0.81	54	0	0.81	22
	BIT	69	0	0.36	69	0	0.36	22
	BYTE	54	1	0.64	54	1	0.64	22
	WORD	54	0	0.97	54	0	0.97	22
	Szukanie wartości zadanej							
	INT	37	0	0.62	37	0	0.62	19
	DINT	41	1	1.38	41	1	1.38	22
	BYTE	35	0	0.46	35	0	0.46	19
	WORD	37	0	0.62	37	0	0.62	19
	Szukanie wartości różnej							
	INT	37	0	0.62	37	0	0.62	19
	DINT	38	0	2.14	38	0	2.14	22
	BYTE	37	0	0.47	37	0	0.47	19
	WORD	37	0	0.62	37	0	0.62	19
	Szukanie wartości większej							
	INT	37	0	1.52	37	0	1.52	19
	DINT	39	0	2.26	39	0	2.26	22
	BYTE	36	1	1.24	36	1	1.24	19
	WORD	37	0	1.52	37	0	1.52	19
	Szukanie wartości większej lub równej							
	INT	37	0	1.48	37	0	1.48	19
	DINT	39	0	2.33	39	0	2.33	22
	BYTE	37	1	1.34	37	1	1.34	19
	WORD	37	0	1.48	37	0	1.48	19

Uwagi:

1. Czasy (w mikrosekundach) dla jednostek centralnych 351 i 352 na podstawie oprogramowania Logicmaster 90-30/20/Micro w wersji 7.
2. W przypadku funkcji wykonujących operacje na tablicach przyrost czasu podany jest w jednostkach o zadanej długości: dla funkcji do operacji bitowych w mikrosekundach/ bit, a dla funkcji do przeprowadzania operacji na danych w mikrosekundach/ liczbę bitów lub słów.
3. Czas w stanie aktywnym podano dla modułów o pojedynczej długości typu %R, %AI i %AQ.
4. Czas dla COMMREQ został zmierzony pomiędzy jednostką centralną a modulem HSC.
5. W przypadku funkcji DOIO jest to czas wysyłania parametrów do modułu z wyjściami dyskretnymi.
6. Jeżeli istnieje więcej niż jedna możliwość, podany czas jest czasem najmniej korzystnym.
7. W przypadku instrukcji dla których w kolumnie Przyrost czasu podano wartość, wartość tę należy pomnożyć przez (Długość - 1), a następnie dodać ten wynik do czasu podstawowego.

Tabela A-2. Czasy wykonywania, modele zaawansowane - kontynuacja

Grupa funkcji	Funkcja	W stanie aktywnym	W stanie nieaktywn.	Przyrost czasu	W stanie aktywnym	W stanie nieaktywn.	Przyrost czasu	Rozm.
		350/351/36x	350/351/36x	350/351/36x	352	352	352	
	Szukanie wartości mniejszej							
	INT	37	0	1.52	37	0	1.52	19
	DINT	41	1	2.27	41	1	2.27	22
	BYTE	37	0	1.41	37	0	1.41	19
	WORD	37	0	1.52	37	0	1.52	19
	Szukanie wartości mniejszej lub równej							
	INT	38	0	1.48	38	0	1.48	19
	DINT	40	1	2.30	40	1	2.30	22
	BYTE	37	0	1.24	37	0	1.24	19
	WORD	38	0	1.48	38	0	1.48	19
Funkcje konwersji	Konwersja na dane typu BCD-4	19	1	–	19	1	–	10
	Konwersja na kod BCD	21	1	–	21	1	–	10
	Konwersja na dane typu REAL	27	0	–	21	0	–	8
	Konwersja na dane typu WORD	28	1	–	30	1	–	11
	Przybliżenie do liczby typu INT	32	0	–	32	0	–	11
	Przybliżenie do liczby typu DINT	63	0	–	31	0	–	11
Funkcje sterujące	Wywołanie podprogramu	72	1	–	73	1	–	7
	Natychmiastowe uaktualnienie stanu wejść/wyjść	114	1	–	115	1	–	13
	Algorytm PID-ISA*	162	34	–	162	34	–	16
	Algorytm PID-IND*	146	34	–	146	34	–	16
	Instrukcje END	–	–	–	–	–	–	–
	Wywoływanie specjalnych funkcji sterownika:							
	#6	22	1	–	22	1	–	10
	# 7 (Odczyt)	75	1	–	75	1	–	10
	# 7 (Ustawianie)	75	1	–	75	1	–	10
	#14	121	1	–	121	1	–	10
	#15	46	1	–	46	1	–	10
	#16	36	1	–	36	1	–	10
	#18	261	1	–	261	1	–	10
	#23	426	0	–	426	0	–	10
	#26//30**	2260	1	–	2260	1	–	10
	#29	20	0	–	20	0	–	10
	#43							
	Para zagnieżdżonych instrukcji MCR/ENDMCR	1	1	–	1	1	–	4
	SER	(Porównać z tabelą A-3)	26.50				(Porównać z tabelą A-3)	

* Czasy dla bloku funkcyjnego PID podano dla jednostki centralnej CPU351, z oprogramowaniem systemowym w. 6.5.

** Czasy dla bloku funkcyjnego #26/30 zmierzono dla modułu licznika impulsów o wysokiej częstotliwości, z 16 wyjściami, zainstalowanego w kasecie z 5-a gniazdami.

Uwagi:

1. Czasy (w mikrosekundach) dla jednostek centralnych 351 i 352 na podstawie oprogramowania Logicmaster 90-30/20/Micro w wersji 7.
2. W przypadku funkcji wykonujących operacje na tablicach przyrost czasu podany jest w jednostkach o zadanej długości: dla funkcji do operacji bitowych w mikrosekundach/bit, a dla funkcji do przeprowadzania operacji na danych w mikrosekundach/ liczbę bitów lub słów.
3. Czas w stanie aktywnym podano dla modułów o pojedynczej długości typu %R, %AI i %AQ.
4. Czas dla COMMREQ został zmierzony pomiędzy jednostką centralną a modulem HSC.
5. W przypadku funkcji DOIO jest to czas wysyłania parametrów do modułu z wyjściami dyskretnymi.
6. Jeżeli istnieje więcej niż jedna możliwość, podany czas jest czasem najmniej korzystnym.
7. W przypadku instrukcji dla których w kolumnie Przyrost czasu podano wartość, wartość tę należy pomnożyć przez (Długość - 1), a następnie dodać ten wynik do czasu podstawowego.

Tabela A-3. Czas wykonywania bloku funkcyjnego SER

Konfigurowanie	Przykład	Czas (μ s)
Sygnal wyjściowy nie wysyłany (stan nieaktywny)	—	26.50
Dla sąsiadujących		
8 kanałów	%I1—8	79.94
16 kanałów	%I1—16	80.58
24 kanałów	%I1—24	81.56
32 kanałów	%I1—32	81.73
8+8 sąsiadujących kanałów	%I1—8 i %Q1—8	111.03
8+8+8 sąsiadujących kanałów	%I1—8, %Q1—8 i %M1—8	143.38
8+8+8+8 sąsiadujących kanałów	%I1—8, %Q1—8 i %M1—8 i %T1—8	175.79
Dla nie sąsiadujących		
8 kanałów	%I1, %M10, %Q3, itp.	299.64
16 kanałów		552.83
24 kanałów		806.35
32 kanałów		1059.85
Kasowanie		
z 8 kanałami	—	162.63
z 16 kanałami	—	267.51
z 24 kanałami	—	372.73
z 32 kanałami	—	477.95

Uwagi: Jeżeli określono gniazdo w module wejść, każdy z czasów (**zarówno dla kanałów sąsiadujących jak i nie sąsiadujących**) należy dodatkowo zwiększyć o 46 μ s.

W momencie wystąpienia wyzwolenia, należy dodatkowo dodać 29 us w przypadku korzystania z formatu BCD lub 148 us w przypadku korzystania z formatu POSIX.

Czasy kasowania podano dla maksymalnej wielkości bufora – 1024 próbek. (Kasowanie powoduje usunięcie wszystkich próbek z bufora próbek).

Zapotrzebowanie instrukcji na pamięć w zaawansowanych modelach jednostek centralnych

Wielkość pamięci to liczba bajtów, zajmowanych przez każdą z instrukcji w drabinie logicznej programu sterującego. Jednostki centralne 351 i 352 wymagają dla większości standardowych funkcji logicznych trzech (3) bajtów pamięci - proszę porównać z Tabelą A3.

Tabela A-4. Zapotrzebowanie instrukcji na pamięć w jednostkach centralnych 350-352, 360, 363 i 364.

Funkcja	Rozm.
Nie aktywna	1
Funkcja logiczna AND języka drabinkowego	1
Funkcja logiczna OR języka drabinkowego	1
Utworzenie duplikatu elementu na samej górze stosu	1
Wypchnięcie jednej pozycji	1
Zainicjowanie stosu	1
Etykieta	5
Skok	5
Wszystkie pozostałe instrukcje	3
Bloki funkcyjne - proszę porównać z Tabelą A-2	–

Czasy wykonywania dla przekaźników i styków

Zamieszczona poniżej tabela podaje czasy wykonywania dla przekaźników i styków w jednostkach centralnych serii 90-30.

Tabela A-5. Czasy wykonywania dla przekaźników i styków

Model jednostki centralnej	Czas wykonywania dla 1000 styków/ przekaźników logicznych
Modele serii 35x i 36x	0,22 ms
Modele 340/ 341	0,3 milisekund
Model 331	0,4 milisekund
Modele 313/ 323	0,6 milisekund
Model 311	18,0 milisekund

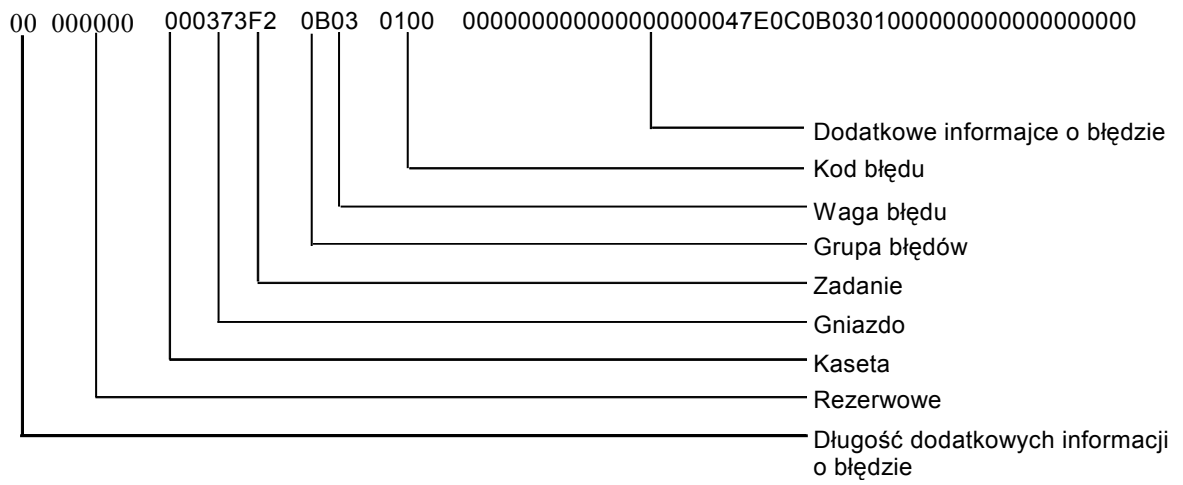
Sterowniki serii 90-30, serii 90-20 i serii 90-Micro posiadają dwie tabele błędów działania: jedna dla błędów działania układów wejść/wyjść, a druga dla wewnętrznych błędów działania sterownika. Informacje zawarte w niniejszym załączniku są pomocne przy analizowaniu formatu komunikatów zapisanych w tabeli błędów działania. Obydwie tabele zawierają podobne dane.

- W tabelach błędów działania sterownika zapisane są:
 - Miejsce wystąpienia błędu
 - Opis błędu
 - Data i czas wystąpienia błędu
- W tabeli błędów działania układów wejść/wyjść zapisane są:
 - Miejsce wystąpienia błędu
 - Adres punktu
 - Kategoria błędu
 - Typ błędu

Tabela błędów działania sterownika

Tabela błędów działania sterownika może być odczytywana za pomocą oprogramowania. Sposób odczytywania tablicy błędów z poziomu oprogramowania opisano w pomocy komputerowej oraz w podręczniku *Logimaster 90 Series 90-30/20/Micro Programming Software User's Manual*, GFK-0466.

Na zamieszczonym poniżej schemacie opisano pola komunikatu "System Configuration Mismatch".



Komunikat System Configuration Mismatch szczegółowo omówiono w poniższej tabeli. (Wszystkie wartości podane są w systemie heksadecymalnym.)

Pole	Wartość	Opis
Długość dodatkowych informacji o błędzie	00	Komunikat ten posiada 8 dodatkowych bajtów o informacji błędzie
Kaseta	00	Kaseta główna (kaseta 0)
Gniazdo	03	Gniazdo 3.
Zadanie	44	
Grupa błędów	0B	Błąd System Configuration Mismatch
Waga błędu	03	Błąd krytyczny
Kod błędu	01	

Poniżej opisano każde z pól tego komunikatu. Podano również dopuszczalne wartości każdego z tych pól.

Długość dodatkowych informacji o błędzie

Bajt ten podaje długość bloku szczegółowych informacji o błędzie, równą 8 lub 24 bajty.

Typ	Kod	Długość dodatkowych informacji o błędzie
Blok krótki	00	8 bajtów
Blok długi	01	24 bajtów

Pole rezerwowe

Tych sześć bajtów ma na celu zapewnienie takiej samej długości tablicy błędów działania sterownika, jak długość tabeli błędów działania układów wejść/wyjść.

Kaseta

Kasety numerowane są od 0 do 7. Numer zero oznacza odwołanie do kasety głównej, zawierającej jednostkę centralną. Kasety 1 do 7 są kasetami rozszerzającymi, podłączanymi do sterownika za pomocą kabla komunikacyjnego.

Gniazdo

Gniazda numerowane są od 0 do 9. Jednostka centralna sterownika zawsze umieszczona jest w gnieździe 1 kasety głównej (kaseta 0).

Zadanie

Zadania numerowane są od 0 do +65 535. Czasami numer zadania wykorzystywany jest przez użytkowników sterowników w celu uzyskania dodatkowych informacji o błędzie.

Grupy błędów

Grupa błędu jest pierwszym kryterium do klasyfikacji błędu. Podaje ona ogólną kategorię do której błąd należy. Tekst objaśniający błąd, wyświetlany przez oprogramowanie Logicmaster 90-30/20/Micro, zależy właśnie od grupy błędu i kodów błędu.

W tabeli B-1 podano wszystkie grupy błędów, które można wyróżnić w działaniu sterownika.

Ostatnia grupa, bez przyporządkowanego numeru, Błąd niesklasyfikowany została zadeklarowana w celu obsługi nowych błędów w systemie, dla których sterownik nie zna kodu. Wszystkie nierozpoznane błędy są klasyfikowane do tej grupy.

Tabela B-1. Grupy błędów działania sterownika:

Numer grupy		Nazwa grupy	Waga błędu
Dziesiąt.	Heksadecym.		
1	1	Loss of, or missing, rack (<i>Brak lub uszkodzenie kasety</i>)	Błąd krytyczny
4	4	Loss of, or missing, option module (<i>Uszkodzenie lub brak modułu wyspecjalizowanego</i>)	Błąd diagnostyczny
5	5	Addition of, or extra, rack (<i>Nie skonfigurowana lub nowa kasety</i>)	Błąd diagnostyczny
8	8	Addition of, or extra, option module (<i>Nie skonfigurowany lub nowy moduł wyspecjalizowany</i>).	Błąd diagnostyczny
11	B	System configuration mismatch (<i>Błędna konfiguracja systemu</i>)	Błąd krytyczny
12	C	System bus error (<i>Błąd magistrali systemowej</i>)	Błąd diagnostyczny
13	D	PLC CPU hardware failure (<i>Uszkodzenie sprzętowe jednostki centralnej sterownika</i>)	Błąd krytyczny
14	E	Non-fatal module hardware failure (<i>Niekrytyczny błąd modułu sprzętowego</i>)	Błąd diagnostyczny
16	10	Option module software failure (<i>Błąd w oprogramowaniu wyspecjalizowanego modułu dodatkowego</i>)	Błąd diagnostyczny
17	11	Program block checksum failure (<i>Błędna suma kontrola programu</i>)	Błąd krytyczny
18	12	Low battery signal (<i>Rozładowane baterie</i>)	Błąd diagnostyczny
19	13	Constant sweep time exceeded (<i>Przekroczony czas trwania cyklu pracy sterownika</i>)	Błąd diagnostyczny
20	14	PLC system fault table full (<i>Przepełnienie tabeli błędów działania sterownika</i>)	Błąd diagnostyczny
21	15	I/O fault table full (<i>Przepełnienie tabeli błędów działania układów wejść/wyjść</i>)	Błąd diagnostyczny
22	16	User Application fault (<i>Błąd w programie sterującym użytkownika</i>)	Błąd diagnostyczny
-	-	Additional PLC fault codes (<i>Błąd nie sklasyfikowany</i>)	Błąd informacyjny
128	80	System bus failure (<i>Uszkodzenie magistrali systemowej</i>)	Błąd krytyczny
129	81	No user's program on power-up (<i>Brak programu sterującego w czasie rozruchu</i>)	Błąd informacyjny
130	82	Corrupted user RAM detected (<i>Wykrycie uszkodzenia pamięci RAM</i>)	Błąd krytyczny
132	84	Password access failure (<i>Niepoprawne hasło dostępu do sterownika</i>)	Błąd informacyjny
135	87	PLC CPU software failure (<i>Błąd oprogramowania jednostki centralnej</i>)	Błąd krytyczny
137	89	PLC sequence-store failure (<i>Błąd w czasie przesyłania danych do sterownika</i>)	Błąd krytyczny

Waga błędu

Można wyróżnić trzy rodzaje wag błędów. Wagi przyporządkowane do błędów w sterownikach serii 90-30 nie mogą być zmieniane przez użytkownika.

Tabela B-2. Tabela wag błędów działania sterownika:

Waga błędu	Działania realizowane przez jednostkę centralną	Kod
Błąd informacyjny	Zarejestrowanie błędu w tabeli błędów	1
Błąd diagnostyczny	Zarejestrowanie błędu w tabeli błędów Ustawienie zmiennych systemowych informujących o wystąpieniu błędu	2
Błąd krytyczny	Zarejestrowanie błędu w tabeli błędów Ustawienie zmiennych systemowych informujących o wystąpieniu błędu Przejsięcie do trybu STOP	3

Kod błędu

Kod błędu zawiera dalsze informacje o błędzie. Każda z grup błędów posiada swój własny zestaw kodów błędów. W Tabeli B-3 zestawiono kody błędów dla grupy Uszkodzone oprogramowanie jednostki centralnej (Grupa 87H)

Tabela B-3. Kody błędów alarmowych dla grupy PLC CPU Software Faults

Dziesiąt.	Heksadecym.	Nazwa
20	14	Corrupted PLC Program Memory
39	27	Corrupted PLC Program Memory
82	52	Backplane Communications Failed
90	5A	User Shut Down Requested
Wszystkie pozostałe		PLC CPU Internal System Error

W Tabeli B-4 zamieszczono kody błędów dla pozostałych grup.

Tabela B-4. Kody błędów alarmowych dla sterownika

Dziesiąt.	Heksadecym.	Nazwa
Kody błędów sterownika dla grupy Loss of Option Module (Brak modułu dodatkowego) (4)		
44	2C	Nieudana próba zresetowania oprogramowania modułu wyspecjalizowanego
45	2D	Nieudana próba zresetowania oprogramowania modułu wyspecjalizowanego
255	FF	Błąd w komunikacji z modułem dodatkowym
79	4F	Uszkodzenie modułu dodatkowego
Kody błędów dla grupy Reset of, Addition of, or Extra Option Module (Ponowne uruchomienie, dodanie modułu lub dodatkowy, wyspecjalizowany moduł (8)		
2	2	Zakończenie ponownego uruchamiania modułu
04	4	Dodanie modułu dodatkowego
05	5	Zresetowanie modułu dodatkowego
	Wszystkie pozostałe	Ponowne uruchomienie, dodanie lub brak konfiguracji dla modułu opcjonalnego
Kody błędów dla grupy Option Module Software Failure (Uszkodzenie oprogramowania modułu dodatkowego (10 heks.))		
1	1	Nieobsługiwany typ płyty
2	2	Przepełnienie skrzynki pocztowej COMMREQ dla komunikatu wyjściowego powodującego wywołanie funkcji COMMREQ
3	3	Skrzynka pocztowa COMMREQ przepełniona w czasie wysyłania odpowiedzi
5	5	Magistrala komunikacyjna sterownika; nie zrealizowane żądanie
11	B	Błąd zasobów (alokacja pamięci, przepełnienie tabeli, itp.)
13	D	Błąd w programie sterującym użytkownika
401	191	Uszkodzenie oprogramowania modułu, żądanie ponownego wczytania
Kody błędów dla grupy System Configuration Mismatch (Błędna konfiguracja systemu) (B heks.)		
8	8	Błąd w konfiguracji dodatkowych modułów analogowych
10	A	Nie obsługiwana funkcja
23	17	Brak pamięci dla programu sterującego
58	3A	Niezgodność konfiguracji modułu dodatkowego
Kody błędów dla grupy System Bus Error (Błąd magistrali systemowej) (heks. C)		
	Wszystkie pozostałe	Błąd magistrali systemowej
Kody błędów dla grupy Program Block Checksum (Błędna suma kontrolna programu sterującego) (heks. 11)		
3	3	Błędna suma kontrolna dla programu lub bloku programu
Kody błędów dla grupy Low Battery Signal (Rozładowane baterie)		
0	0	Uszkodzone baterie jednostki centralnej sterownika lub innego modułu
1	1	Rozładowane baterie jednostki centralnej sterownika lub innego modułu
Kody błędów dla grupy Application Fault (Uszkodzenie programu sterującego użytkownika) (heks. 16)		
2	2	Przekroczenie czasu przez zegar wyłączający sterownika
5	5	Dana instrukcja nie może pracować w trybie COMMREQ - WAIT
6	6	COMMREQ –Niewłaściwy identyfikator ID zadania
7	7	Przepełnienie stosu aplikacji
Kody błędów dla grupy System Bus Failure Group (Błąd magistrali systemowej) (heks. 80)		
1	1	System operacyjny
Kody błędów dla grupy Corrupted User RAM on Powerup (Wykrycie w czasie rozruchu uszkodzenia pamięci RAM użytkownika) (heks. 82)		
1	1	Uszkodzenie pamięci RAM użytkownika w czasie rozruchu
2	2	Wykryty niewłaściwy kod logiczny Opcode
3	3	PLC_ISCP_PC_OVERFLOW
4	4	PRG_SYNTAX_ERR
Kody błędów dla grupy PLC CPU Hardware Faults (Uszkodzenie sprzętowe jednostki centralnej sterownika) (heks. D)		
	Wszystkie pozostałe	Uszkodzenie sprzętowe jednostki centralnej

Dodatkowe informacje o błędzie

W polu tym zapisane są szczegółowe informacje o błędzie. Przykładowo, mogą to być:

Grupa Uszkodzona pamięć RAM użytkownika: Dla czterech kodów błędów, z grupy Błędna konfiguracja systemu podawane są dodatkowe informacje:

Tabela B-5. Informacje o błędzie działania sterownika - Wykryty niewłaściwy kod logiczny Opcode

Dodatkowe informacje o błędzie	Niewłaściwy numer modelu
[0]	Zawartość rejestru błędów ISCP
[1]	Niewłaściwy kod OPCODE
[2,3]	Licznik programu ISCP
[4,5]	Numer funkcji

W przypadku wykrycia uszkodzenia pamięci RAM jednostki centralnej sterownika (jedno z pierwszych uszkodzeń, podawanych w grupie Uszkodzenia sprzętowe jednostki centralnej sterownika), w pierwszych czterech bajtach tego pola zapamiętywany jest adres wystąpienia uszkodzenia.

Uszkodzenie sprzętowe jednostki centralnej (uszkodzenie pamięci RAM):

Czas i data wystąpienia błędu działania sterownika

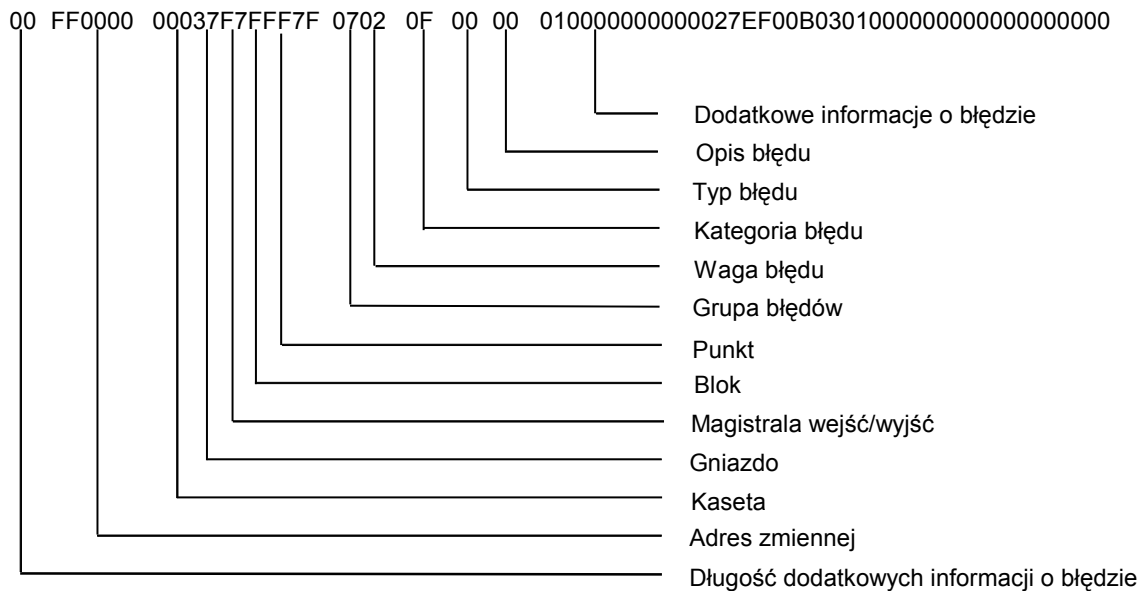
Czas i data wystąpienia błędu działania sterownika zapisywane są w sześciu bajtach. (w systemie heksadecymalnym.)

Tabela B-6. Czas i data wystąpienia błędu działania sterownika

Numer bajtu	Opis
1	Sekundy
2	Minuty
3	Godziny
4	Dzień miesiąca
5	Miesiąc
6	Rok

Tabela błędów działania układów wejść/wyjść

Na zamieszczonym poniżej schemacie pokazano wartości heksadecymalne dla każdego z pól komunikatu o błędzie.



Poniżej opisano każde z pól tego komunikatu. Podano również dopuszczalne wartości każdego z tych pól.

Długość dodatkowych informacji o błędzie

Bajt ten informuje o długości bloku dodatkowych informacji o błędzie: 5 bajtów lub 21 bajty.

Tabela B-7. Długość bloku dodatkowych informacji o błędzie

Typ	Kod	Długość bloku dodatkowych informacji o błędzie
Blok krótki	02	5 bajtów
Blok długi	03	21 bajtów

Adres zmiennej:

Adres zmiennej zajmuje trzy bajty i określa typ pamięci oraz adres względny (lub przemieszczenia) punktu, dla którego stwierdzony został błąd. Jeżeli uszkodzenie wystąpiło w bloku Genius, adres ten odnosi się do pierwszego punktu bloku, w którym stwierdzono uszkodzenie.

Tabela B-8. Adres zmiennej /wyjść

Bajt	Opis	Zakres wartości
0	Typ pamięci	0 –FF
1–2	Adres względny	0 –7FF

Bajt określający typ pamięci może przyjmować wartości podane w zamieszczonej poniżej tabeli.

Tabela B-9. Kodowanie typów pamięci

Typ pamięci	Wartość (Heksadecymalnie)
Wejście analogowe	0A
Wyjście analogowe	0C
Zgrupowane wejścia analogowe	0D
Wejście dyskretne	10 lub 46
Wyjście dyskretne	12 lub 48
Zgrupowane wyjścia dyskretne	1F

Adres wystąpienia błędu działania układów wejść/wyjść.

Adres wystąpienia błędu działania układów wejść/wyjść jest adresem o długości sześciu bajtów określającym kasetę, gniazdo, magistralę, blok i adres punktu dla którego stwierdzone zostało uszkodzenie. Adres punktu jest słowem, wszystkie pozostałe adresy mają długość jednego bajtu. Przy określaniu adresu wystąpienia błędu, nie muszą być podawane wszystkie pięć wartości.

Jeżeli adres wystąpienia błędu w działaniu układów wejść/wyjść nie zawiera wszystkich pięciu adresów, miejsce do którego podawane są prawidłowe wartości oznaczone jest przez wprowadzenie liczby heksadecymalnej 7F. Przykładowo, jeżeli bajt magistrali ma wartość 7F, zlokalizowano uszkodzenie w module. Podawane są więc wyłącznie poprawne wartości dla kasety i gniazda.

Kaseta

Kasety numerowane są od 0 do 7. Kaseta główna ma numer zero, zawiera ona jednostkę centralną. Kasety o numerach 1 do 7 są kasetami rozszerzającymi.

Gniazdo

Gniazda numerowane są od 0 do 9. Jednostka centralna sterownika zawsze umieszczona jest w gnieździe 1 kasety główna (kaseta 0).

Punkt

Punkty mogą mieć przyporządkowane numery od 1 do 1024 (w systemie dziesiętnym). Podana wartość informuje który z punktów w bloku został uszkodzony, o ile jest to jest to uszkodzenie punktu.

Grupy błędów działania układów wejść/wyjść

Grupa błędu jest pierwszym kryterium do klasyfikacji błędu. Podaje ona ogólną kategorię do której błąd należy. Tekst objaśniający błąd, wyświetlany przez oprogramowanie Logicmaster 90-30/20/Micro, zależy właśnie od grupy błędu i kodu błędu.

W tabeli B-10 podano wszystkie grupy błędów działania układów wejść/wyjść. Grupy o numerach mniejszych od 80 (heksadecymalnie) zawierają znane błędy.

Ostatnia grupa, bez przyporządkowanego numeru, **Nie sklasyfikowany błąd działania sterownika** została zadeklarowana w celu obsługi nowych błędów w systemie, dla których sterownik nie zna kodu. Wszystkie nierozpoznane błędy są klasyfikowane do tej grupy.

Tabela B-10. Grupy błędów działania układów wejść/wyjść

Numer grupy	Nazwa grupy	Waga błędu
3	Loss of, or missing, I/O module (Uszkodzenie lub brak modułu wejść/wyjść)	Błąd diagnostyczny
7	Addition of, or extra, I/O module (Nie skonfigurowany lub nowy moduł wejść/wyjść)	Błąd diagnostyczny
9	IOC or I/O bus fault (Uszkodzenie kontrolera magistrali lub magistrali wejść/wyjść)	Błąd diagnostyczny
A	I/O module fault (Uszkodzenie modułu wejść/wyjść)	Błąd diagnostyczny
–	Additional I/O fault codes (Nie sklasyfikowany błąd działania układów wejść/wyjść)	Zgodnie z informacjami

Wagi błędów działania układów wejść/wyjść

Zakres działań podejmowanych w przypadku wystąpienia błędu zależy od wagi tego błędu. Wagi błędów wymieniono w Tablicy B-11.

Tabela B-11. Wagi błędów działania układów wejść/wyjść

Waga błędu	Działania realizowane przez jednostkę centralną	Kod
Błąd informacyjny	Zarejestrowanie błędu w tabeli błędów	1
Błąd diagnostyczny	Zarejestrowanie błędu w tabeli błędów Ustawienie zmiennych systemowych informujących o wystąpieniu błędu	2
Błąd krytyczny	Zarejestrowanie błędu w tabeli błędów Ustawienie zmiennych systemowych informujących o wystąpieniu błędu Przejsście do trybu STOP	3

Dodatkowe informacje o błędach działania układów wejść/wyjść

Każda z tabel błędów działania układów wejść/wyjść może zawierać do 5 bajtów dodatkowych informacji o zaistniałym błędzie.

Symboliczne informacje o błędach

W tabeli B-12 wymieniono dane, wymagane do konfigurowania obwodu bloku.

Tabela B-12. Blok szczegółowych informacji o błędach działania układów wejść/wyjść

Numer dziesiętny	Heksadecymalnie	Opis
<i>Konfiguracja obwodu</i>		
	1	Obwód jest wejściem – trzystanowym
	2	Obwód jest wejściem
	3	Obwód jest wyjściem

Wagi błędów specyficznych

Wymuszenie obwodów jest zapisywane do tablicy błędów jako błąd informacyjny. Wszystkie pozostałe błędy są błędami diagnostycznymi lub krytycznymi.

Błędy: niezgodność numeru modułu, niezgodność typu wejść/wyjść oraz brak modułów wejść/wyjść są zapisywane w tabeli błędów działania sterownika, w grupie Błędna konfiguracja systemu. Nie są one zapisywane w tabeli błędów działania układów wejść/wyjść.

Czas zarejestrowania błędu

Czas i data wystąpienia błędu działania sterownika zapisywane są w sześciu bajtach. Stosowany jest format zapisu BCD.

Tabela B-13. Czas i data wystąpienia błędu działania układów wejść/wyjść

Numer bajtu	Opis
1	Sekundy
2	Minuty
3	Godziny
4	Dzień miesiąca
5	Miesiąc
6	Rok

Wprowadzenie znaku & i następnie nazwy mnemonicznej instrukcji umożliwia szybkie wpisanie czy odszukanie instrukcji w czasie edycji programu sterującego. W przypadku niektórych instrukcji, można również podać adres zmiennej lub nazwę pomocniczą, etykietę lub adres położenia zmiennej.

W niniejszym załączniku wyszczególniono nazwy mnemoniczne dla instrukcji oprogramowania Logicmaster 90-30/20/Micro. Pełne nazwy mnemoniczne podane są w 3 kolumnie tej tabeli, natomiast najkrótsze nazwy, które mogą być wprowadzane dla każdej z instrukcji podano w kolumnie 4.

W dowolnym momencie pracy z oprogramowaniem można wyświetlić ekran z pomocą, zawierający te nazwy mnemoniczne. Należy w tym celu wcisnąć klawisze ALT-I.

Grupa funkcji	Instrukcja	Nazwa mnemoniczna						
		Wszystkie	INT	DINT	BIT	BYTE	WORD	REAL
Styki	Dowolny styk	&CON	&CON					
	Styk otwarty	&NOCON	&NOCON					
	Styk zamknięty	&NCCON	&NCCON					
	Styk kontynuacji	&CONC	&CONC					
Przełączniki	Dowolny przełącznik	&COI	&COI					
	Przełącznik o stykach otwartych	&NOCOI	&NOCOI					
	Przełącznik o stykach zamkniętych	&NCCOI	&NCCOI					
	Przełącznik uaktywniany zboczem narastającym sygnału	&PCOI	&PCOI					
	Przełącznik uaktywniany zboczem opadającym sygnału	&NCOI	&NCOI					
	Przełącznik SET	&SL	&SL					
	Przełącznik RESET	&RL	&RL					
	Przełącznik SET z pamięcią	&SM	&SM					
	Przełącznik RESET z pamięcią	&RM	&RM					
	Przełącznik o stykach otwartych z pamięcią	&NOM	&NOM					
	Przełącznik o stykach zamkniętych z pamięcią	&NCM	&NCM					
	Przełącznik kontynuacji	&COILC	&COILC					
	Szyny	Poziome	&HO	&HO				
Pionowe		&VE	&VE					
Przełączniki czasowe	Przełącznik czasowy włączający z pamięcią	&ON	&ON					
	Przełącznik czasowy włączający	&TM	&TM					
	Przełącznik czasowy wyłączający	&OF	&OF					
Liczniki	Licznik zliczający w górę	&UP	&UP					
	Licznik zliczający w dół	&DN	&DN					

Grupa funkcji	Instrukcja	Nazwa mnemoniczna							
		Wszystkie	BCD-4	INT	DINT	BIT	BYTE	WORD	REAL
Funkcje matematycz.	Dodawanie	&AD		&AD_I	&AD_DI				&AD_R
	Odejmowanie	&SUB		&SUB_I	&SUB_DI				&SUB_R
	Mnożenie	&MUL		&MUL_I	&MUL_DI				&MUL_R
	Dzielenie bez reszty	&DIV		&DIV_I	&DIV_DI				&DIV_R
	Dzielenie modulo	&MOD		&MOD_I	&MOD_DI				&MOD_R&SQ_R
	Pierwiastek kwadratowy	&SQ		&SQ_I	&SQ_DI				
	Sinus	&SIN							
	Cosinus	&COS							
	Tangens	&TAN							
	Arcus sinus	&ASIN							
	Arcus cosinus	&ACOS							
	Arcus tangens	&ATAN							
	Logarytm dziesiętny	&LOG							
	Logarytm naturalny	&LN							
	Potęga liczby e	&EXP							
Potęga liczby x	&EXPT								
Relacje	Równy	&EQ		&EQ_I	&EQ_DI				&EQ_R
	Różny	&NE		&NE_I	&NE_DI				&NE_R
	Większy	>		>_I	>_DI				>_R
	Większy lub równy	&GE		&GE_I	&GE_DI				&GE_R
	Mniejszy	<		<_I	<_DI				<_R
	Mniejszy lub równy	&LE		&LE_I	&LE_DI				&LE_R
Operacje bitowe	AND	&AN						&AN_W	
	OR	&OR						&OR_W	
	XOR	&XO						&XO_W	
	NOT	&NOT						&NOT_W	
	Przesunięcie słowa bitowego w lewo	&SHL						&SHL_W	
	Przesunięcie słowa bitowego w prawo	&SHR						&SHR_W	
	Przesunięcie słowa bitowego w lewo w obiegu zamkniętym	&ROL						&ROL_W	
	Przesunięcie słowa bitowego w prawo w obiegu zamkniętym	&ROR						&ROR_W	
	Sprawdzanie wartości pojedynczego bitu	&BS						&BS_W	
	Ustawianie wartości pojedynczego bitu na 1	&BCL						&BCL_W	
	Ustawianie wartości pojedynczego bitu na 0	&BP						&BP_W	
	Lokalizowanie pierwszego bitu o wartości 1	&MCMP						&MCM_W	
	Porównanie z maskowaniem								
	Funkcje konwersji	Konwersja na INT	&TO_INT	&TO_INT_BCD4	&MOV				
Konwersja na DINT		&TO_DINT		&BLKM					
Konwersja na BCD-4		&BCD4		&BLKC					&BCD4_R
Konwersja na REAL		&TO_REAL		&SHF	&TO_REAL_DI			&TO_REAL_W	
Konwersja na WORD		&TO_W		&BI					
Przybliżenie do liczby typu INT		&TRINT		&COMMR					
Przybliżenie do liczby typu DINT		&TRDINT							

Grupa funkcji	Instrukcja	Nazwa mnemoniczna						
		Wszystkie	INT	DINT	BIT	BYTE	WORD	REAL
Przemieszczenie danych	Przemieszczenie	&MOV	&MOV_I		&MOV_BI		&MOV_W	&MOV_R
	Przemieszczanie bloku	&BLKM	&BLKM_I				&BLKM_W	&BLKM_R
	Zerowanie bloku pamięci	&BLKC						
	Rejestr przemieszczający	&SHF			&SHF_BI		&AR_W	
	Przemieszczanie jedynki	&BI						
	Obsługa funkcji specjalnych sterownika	&COMMR						
Operacje	Kopiowanie danych	&AR	&AR_I	&AR_DI	&AR_BI	&AR_BY	&AR_W	
	Szukanie wartości zadanej	&SRCHE	&SRCHE_I	&SRCHE_DI		&SRCHE_BY	&SRCHE_W	
	Szukanie wartości różnej	&SRCHN	&SRCHN_I	&SRCHN_DI		&SRCHN_BY	&SRCHN_W	
	Szukanie wartości większej	&SRCHGT	&SRCHGT_I	&SRCHGT_DI		&SRCHGT_BY	&SRCHGT_W	
	Szukanie wartości większej lub równej	&SRCHGE	&SRCHGE_I	&SRCHGE_DI		&SRCHGE_BY	&SRCHGE_W	
	Szukanie wartości mniejszej	&SRCHLT	&SRCHLT_I	&SRCHLT_DI		&SRCHLT_BY	&SRCHLT_W	
	Szukanie wartości mniejszej lub równej	&SRCHLE	&SRCHLE_I	&SRCHLE_DI		&SRCHLE_BY	&SRCHLE_W	
Funkcje sterujące	Wywołanie podprogramu	&CA						
	Natychmiastowe uaktualnienie stanu wejść/wyjść	&DO						
	SER	&SER						
	Algorytm PID-ISA	&PIDIS						
	Algorytm PID-IND	&PIDIN						
	Zerowanie SFC	&SFCR						
	End	&END						
	Komentarz	&COMME						
	Funkcje systemowe	&SV						
	MCR	&MCR						
	MCRN	&ENDMCR						
	MCRN	&MCRN						
	ENDMCRN	&ENDMCRN						
	Skok	&JUMP						
	Zagnieżdżona instrukcja JUMP	&JUMPN						
	Etykieta	&LABEL						
Zagnieżdżona etykieta	&LABELN							

W niniejszym załączniku opisano skróty klawiszowe wykorzystywane w oprogramowaniu. Podane tu informacje mogą zostać również wyświetlone na ekranie. Należy w tym celu wcisnąć klawisze ALT-K.

Skrót klawiaturowy	Opis	Skróty klawiaturowe	Opis
<i>Skróty klawiaturowe globalnet</i>			
ALT-A	Przerwij	CTRL-Break	Zamknięcie programu
ALT-C	Wymazanie zawartości pola	Esc	Powiększenie
ALT-M	Zmiana trybu programatora	CTRL-Home	Przywołanie poprzedniej linii poleceń
ALT-R	Zmiana trybu pracy sterownika (Run/Stop)	CTRL-End	Przywołanie następnej linii poleceń
ALT-E	Wyświetlenie/ ukrycie paska statusu	CTRL- ←	Przesunięcie kursora w lewo w obrębie pola
ALT-J	Wyświetlenie/ ukrycie linii poleceń	CTRL- →	Przesunięcie kursora w prawo w obrębie pola
ALT-L	Wyświetlenie zawartości kartoteki	CTRL-D	Zmniejszenie adresu zmiennej
ALT-P	Wydruk ekranu	CTRL-U	Zwiększenie adresu zmiennej
ALT-H	Pomoc	Tabulator	Zmiana/ zwiększenie wartości pola
ALT-K	Pomoc na temat skrótów klawiszowych	Shift-Tab	Zmiana/ zmniejszenie wartości pola
ALT-I	Pomoc na temat nazw mnemonicicznych	Enter	Zaakceptowanie wartości wprowadzonej w polu
ALT-N	Przełączenie opcji wyświetlania	CTRL-E	Wyświetlenie ostatniego błędu systemowego
ALT-T	Włączenie trybu uczenia (Teach)	F12 lub szary minus (-)	Zmiana wartości zmiennej dyskretnej
ALT-Q	Wyłączenie trybu uczenia (Teach)	F11 lub szary klawisz mnożenia (*)	Wymuszona zmiana wartości zmiennej dyskretnej
ALT-N	Odtworzenie pliku n (n=0 do 9)		
<i>Skróty klawiszowe dostępne wyłącznie w edytorze programu</i>			
ALT-B	Włączenie/ wyłączenie sygnału dźwiękowego edytora	Szary klawisz dodawania (+)	Akceptacja szczebla
ALT-D	Usunięcie elementu szczebla/ usunięcia szczebla	Enter	Akceptacja szczebla
ALT-S	Zapisanie bloku w sterowniku i na dysku	CTRL-PgUp	Poprzedni szczebel
ALT-X	Wyświetlenie współczynnika powiększenia	CTRL-PgDn	Następny szczebel
ALT-U	Uaktualnienie pliku na dysku		Szyna pozioma
ALT-V	Wyświetlenie okna z tabelą zmiennych		Szyna pionowa
ALT-F2	Przejdźcie do tabeli zmiennych	Tabulator	Przejdźcie do następnego parametru
<i>Specjalne skróty klawiszowe</i>			
ALT-O	Zmiana hasła. Skróót dostępny wyłącznie w oknie dialogowym do zmiany hasła w oprogramowaniu do konfiguracji.		

Karta z pomocą zamieszczona na następnej stronie zawiera skróty klawiszowe i nazwy mnemoniczne dla oprogramowania Logicmaster 90-30/20/Micro. Karta ta sporządzona jest w trzech kopiach oraz posiada perforowane krawędzie, co ułatwia odłączenie jej od podręcznika.

Na tej stronie należy wydrukować stronę 1 GFJ-055D.

Na tej stronie należy wydrukować stronę 2 GFJ-055D.

Przy korzystaniu z liczb zmiennoprzecinkowych należy pamiętać o kilku zasadach. W pierwszym punkcie omówiono zagadnienia ogólne. Instrukcje do wprowadzania i wyświetlania liczb zmiennoprzecinkowych podano na stronie E-5 i stronach następnych.

Uwaga

Operacje zmiennoprzecinkowe mogą być realizowane **wyłącznie** w jednostkach centralnych 35x i 36x z oprogramowaniem systemowym ver. 9 lub nowsza oraz we wszystkich wersjach jednostek centralnych CPU352.

Informacje ogólne

Oprogramowanie pozwala na edycję, wyświetlanie, zapisywanie oraz przeglądanie liczb zmiennoprzecinkowych. Liczby zmiennoprzecinkowe są również parametrami niektórych funkcji. Należy jednak pamiętać, że warunkiem korzystania w oprogramowaniu z liczb zmiennoprzecinkowych jest posiadanie odpowiedniej jednostki centralnej (proszę porównać z zamieszczoną powyżej uwagą). Liczby zmiennoprzecinkowe reprezentowane są w notacji dziesiętnej, z wyświetlaniem sześciu znaczących cyfr.

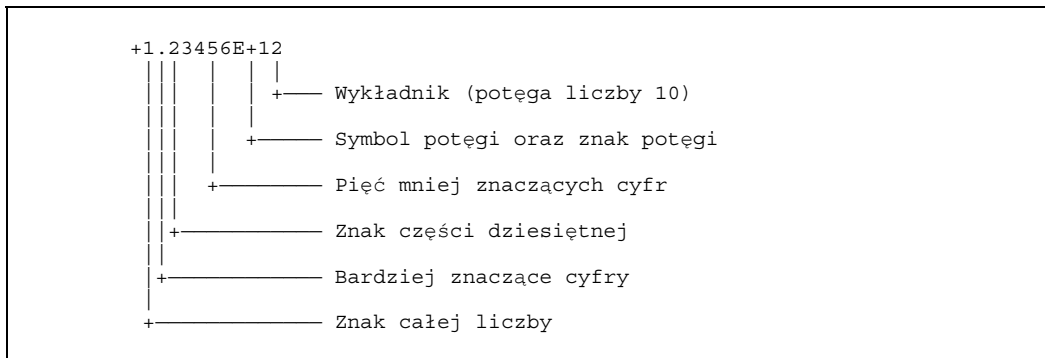
Uwaga

Terminy "liczby zmiennoprzecinkowe" i "liczby rzeczywiste" są w niniejszym podręczniku używane wymiennie, oznaczają ten sam typ danych.

Stosowany jest opisany poniżej format. Dla liczb z zakresu 9 9999999 do 0.0001 na wyświetlaczu nie jest pokazywany wykładnik, a liczba cyfr znaczących wynosi sześć lub siedem. Przykładowo:

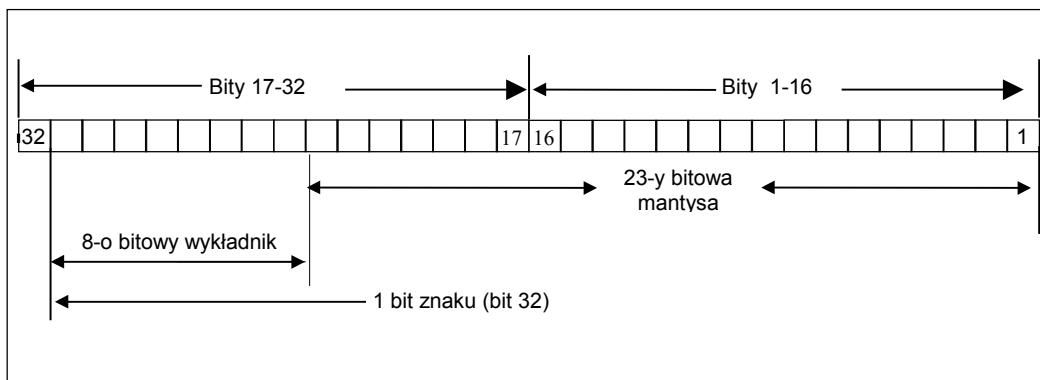
Wartość	Format wyświetlania	Opis
.000123456789	+0.0001234567	Dziesięć cyfr, sześć lub siedem cyfr znaczących
-12.345e-2	-.1234500	Siedem cyfr, sześć lub siedem cyfr znaczących
1234	+1234.000	Siedem cyfr, sześć lub siedem cyfr znaczących

Jeżeli wartość nie mieści się w podanym powyżej przedziale, wyświetlanych jest jedynie sześć cyfr znaczących, w następującym formacie:

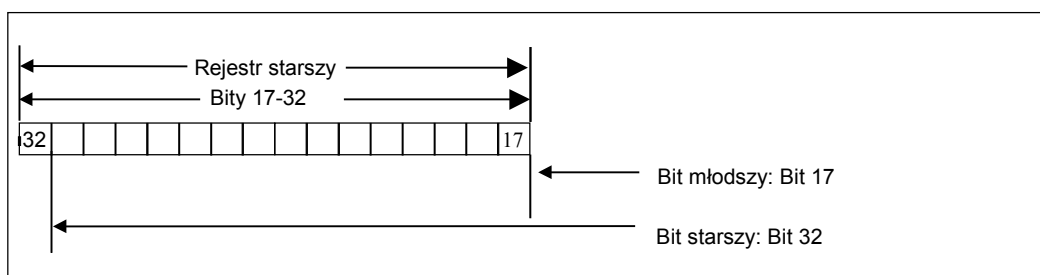
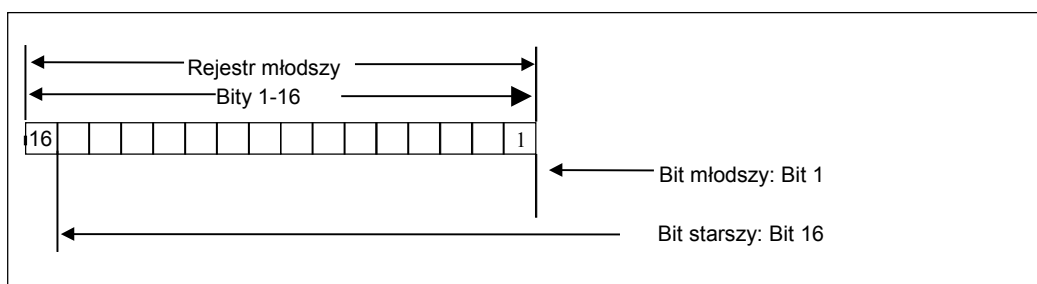


Format wewnętrznej reprezentacji liczb zmiennoprzecinkowych

Liczby zmiennoprzecinkowe pamiętane są w standardowym formacie IEEE pojedynczej precyzji. Do reprezentacji tych liczb wymagane są 32 bity, czyli dwa (sąsiednie) 16 bitowe rejestry sterownika. Sposób kodowania tych bitów podano na zamieszczonym poniżej schemacie.



Kolejny schemat pokazuje wykorzystanie rejestrów przez liczbę zmiennoprzecinkowa pojedynczej precyzji. Na schemacie tym, jeżeli przykładowo liczba zmiennoprzecinkowa zajmuje rejestry R5 i R6, rejestr R5 jest rejestrem mniej znaczącym, a rejestr R6 rejestrem bardziej znaczącym.



Wartości liczb zmiennoprzecinkowych

Zamieszczona poniżej tabela umożliwia obliczenie wartości liczby zmiennoprzecinkowej na podstawie zawartości dwóch sąsiadujących ze sobą rejestrów.

Wykładnik (e)	Mantysa (f)	Wartość liczby zmiennoprzecinkowej
255	Różna od zera	Niepoprawna wartość
255	0	$-1^s * \infty$
$0 < e < 255$	Dowolna wartość	$-1^s * 2^{e-127} * 1.f$
0	Różna od zera	$-1^s * 2^{-126} * 0.f$
0	0	0

f = Mantysa: Mantysa jest funkcją binarną.

e = Wykładnik. Wykładnik jest liczbą całkowitą E taką że: $E+127$ jest potęgą liczby 2 przez którą musi być pomnożona mantysa w celu otrzymania wartości zmiennoprzecinkowej.

s = Bit znaku.

* = Operator mnożenia.

Przykładowo, zanalizowana zostanie reprezentacja liczby zmiennoprzecinkowej 12.5. Reprezentacja zmiennoprzecinkowa tej wartości, zgodnie ze standardem IEEE, wygląda następująco:

01000001 01001000 00000000 00000000

lub 41480000 (heksadecymalnie). n najbardziej znaczący bit (bit znaku) jest równy zero (s=0).

Następnych osiem najbardziej znaczących bitów jest równe 10000010, czyli 130 w systemie dziesiętnym (e=130).

Mantysa pamiętana jest jako dziesiętna liczba binarna, z kropką dziesiętną poprzedzającą najbardziej znaczące 23 bity. Tak więc, najbardziej znaczący bit w mantysie jest wielokrotnością 2^{-1} , następny najbardziej znaczący bit jest wielokrotnością 2^{-2} , itd., do najmniej znaczącego bitu, który jest wielokrotnością 2^{-23} . Ostatnie 23 bity (mantysa) to:

1001000 00000000 00000000

Tak więc wartość mantysy wynosi w takim przypadku $.5625$ (tzn. $2^{-1} + 2^{-4}$).

Ponieważ $e > 0$ i $e < 255$, zastosowana zostanie trzecia formuła z podanej powyżej tabeli:

$$\begin{aligned}
 \text{wartość} &= -1^s * 2^{e-127} * 1.f \\
 &= -1^0 * 2^{130-127} * 1.5625 \\
 &= 1 * 2^3 * 1.5625 \\
 &= 8 * 1.5625 \\
 &= 12.5
 \end{aligned}$$

przedstawiona powyżej analiza udowadnia poprawność tej reprezentacji binarnej.

Zakres wartości, które mogą być reprezentowane w ten sposób wynosi : $\pm 1.401298E-45$ do $\pm 3.402823E+38$.

Wprowadzanie i wyświetlanie liczb zmiennoprzecinkowych

Dla mantysy można wprowadzić i zapamiętać sześć lub siedem znaczących cyfr dokładności, nie mniej jednak oprogramowanie wyświetla wyłącznie pierwszych sześć cyfr mantysy. Mantysa może być poprzedzona znakiem dodatnim lub ujemnym. Jeżeli znak nie zostanie wprowadzony, przyjmowane jest domyślnie, że jest to znak dodatni.

Po wykładniku należy wprowadzić literę E lub e, a mantysa musi zawierać kropkę dziesiętną co pozwoli na uniknięcie pomylenia jej z liczbą heksadecymalną. Należy również podać znak wykładnika, jeżeli jednak nie zostanie on podany, domyślnie przyjmowane jest, że jest on dodatni. Jeżeli wykładnik nie zostanie wprowadzony, przyjmuje się domyślnie, że jest on równy zero. Nie dopuszcza się stosowania znaku spacji w liczbach zmiennoprzecinkowych.

W celu ułatwienia wprowadzania, zarówno w linii poleceń jak i w polach danych dopuszczalne jest stosowanie kilku różnych formatów. Dopuszczalne jest między innymi wprowadzanie liczby całkowitej, liczby dziesiętnej oraz liczby dziesiętnej z wykładnikiem. Tak wprowadzone liczby są konwertowane na standardowy format wyświetlania, natychmiast po wprowadzeniu danych i wciśnięciu klawisza Enter.

Poniżej podano przykłady poprawnie wprowadzonych wartości zmiennoprzecinkowych oraz format ich standardowego wyświetlania.

Wartość	Format wyświetlania
250	+250,0000
+4	+4.000000
-2383019	-2383019.
34.	+34.00000
-.0036209	-.003620900
12.E+9	+1.20000E+10
-.0004E-11	-4.00000E-15
731.0388	+731.0388
99.20003e-29	+9.92000E-28

Następna tabela zawiera przykłady błędnie wprowadzonych wartości zmiennoprzecinkowych.

Błędnie wprowadzona wartość	Objaśnienia
-433E23	Brak kropki dziesiętnej.
10e-19	Brak kropki dziesiętnej.
10.e19	Pomiędzy cyfry i znaki mantysy nie można wprowadzać spacji. W tym przypadku przyjęta zostanie wartość 10.e0 oraz wyświetlony zostanie komunikat o błędzie.
4.1e19	Pomiędzy cyfry i znaki wykładnika nie można wprowadzać spacji. W tym przypadku przyjęta zostanie wartość 4.1e0 oraz wyświetlony zostanie komunikat o błędzie.

Błędy w operacjach i liczbach zmiennoprzecinkowych

W przypadku jeżeli w jednostce centralnej 352, w wyniku działania funkcji operującej na danych typu REAL otrzymana zostanie wartości większa od 3.402823E+38 lub mniejsza od -3.402823E+38 generowany jest błąd. W przypadku pozostałych sterowników serii 90-30, zakres dopuszczalnych wartości liczb zmiennoprzecinkowych jest większy od 2^{16} lub mniejszy od -2^{16} . Jeżeli wartość nie mieści się w tym przedziale, nie jest wysyłany sygnał wyjściowy OK, a wynik w takiej sytuacji ma dodatnią, wartość nieskończenie dużą, tzn. większą od 3.402823E+38 w przypadku jednostki centralnej CPU352 lub większą od 2^{16} w przypadku pozostałych modeli) lub też ma wartość ujemną, nieskończenie dużą (dla wartości mniejszych od -3.402823E+38 lub od -2^{16} dla pozostałych modeli). Wystąpienie takiej sytuacji może zostać stwierdzone poprzez zbadanie wartości sygnału wyjściowego OK.

POS_INF	= 7F800000h	Wartość dodatnia, nieskończenie duża IEEE, w systemie heksadecymalnym.
NEG_INF	= FF800000h	Wartość ujemna, nieskończenie duża IEEE, w systemie heksadecymalnym.

Uwaga

Przy korzystaniu z oprogramowania wykonującego operacje na liczbach zmiennoprzecinkowych (wszystkie modele w których można korzystać z liczb zmiennoprzecinkowych za wyjątkiem jednostki centralnej 352), wartości są zaokrąglane do zera (0) dla $\pm 1.175494E-38$.

Jeżeli otrzymane wartości nieskończenie duże zostaną wykorzystane jako parametr innej funkcji operującej na danych typu REAL, może to być powodem nieokreślonego wyniku. Ten nieokreślony wynik określany jest terminem NaN (Not a Number - Niepoprawna liczba). Przykładowo, wynik dodawania dodatniej wartości nieskończenie dużej do ujemnej wartości nieskończenie dużej jest nieokreślony. Jeżeli więc parametrami wywołania funkcji ADD_REAL są dodatnia wartość nieskończenie duża i ujemna wartość nieskończenie duża, otrzymany wynik będzie równy NaN.

W przypadku jednostki centralnej 352, każda funkcja operująca na danych typu REAL, generuje swoją wartość NaN, identyfikującą jednoznacznie funkcję.

NaN_SW	= FFFFFFFFh	Progama wartość NaN
NaN_ADD.	= 7F81FFFFh	Wartość heksadecymalna dla błędu dodawania liczb rzeczywistych.
NaN_SUB	= 7F81FFFFh	Wartość heksadecymalna dla błędu odejmowania liczb rzeczywistych.
NaN_MUL	= 7F82FFFFh	Wartość heksadecymalna dla błędu mnożenia liczb rzeczywistych
NaN_DIV	= 7F83FFFFh	Wartość heksadecymalna dla błędu dzielenia liczb rzeczywistych
NaN_SQRT	= 7F84FFFFh	Wartość heksadecymalna dla błędu obliczania pierwiastka kwadratowego dla liczby rzeczywistej.
NaN_LOG	= 7F85FFFFh	Wartość heksadecymalna dla błędu obliczania logarytmu z liczby rzeczywistej.
NaN_POW0	= 7F86FFFFh	Wartość heksadecymalna dla błędu obliczania potęgi liczby rzeczywistej.
NaN_SIN	= 7F87FFFFh	Wartość heksadecymalna dla błędu obliczania sinusa liczby rzeczywistej.
NaN_COS	= 7F88FFFFh	Wartość heksadecymalna dla błędu obliczania cosinusa liczby rzeczywistej.
NaN_TAN	= 7F89FFFFh	Wartość heksadecymalna dla błędu obliczania tangensa liczby rzeczywistej.
NaN_ASIN	= 7F8AFFFFh	Wartość heksadecymalna dla błędu obliczania arcus sinusa liczby rzeczywistej.
NaN_ACOS	= 7F8BFFFFh	Wartość heksadecymalna dla błędu obliczania arcus cosinusa liczby rzeczywistej.
NaN_BCD	= 7F8CFFFFh	Błąd w czasie konwersji liczby w formacie BCD-4 na liczbę rzeczywistą
REAL_INDEF	= FFC00000h	Nieokreślona liczba rzeczywista, błąd spowodowany próbą dzielenia przez 0.

Wszystkie pozostałe jednostki centralne, dysponujące możliwością operowania na liczbach rzeczywistych, dają jedną (1) wartość NaN: FFFF FFFF.

Jeżeli wynik równy NaN zostanie przesłany jako parametr do innej funkcji, wynik tej funkcji również będzie równy NaN (niepoprawna wartość). Przykładowo, jeżeli wartość NaN_ADD zostanie wykorzystana jako pierwszy parametr funkcji SUB_REAL, wynik funkcji SUB_REAL będzie równy NaN_ADD. Jeżeli obydwa parametry będą równe NaN, wynik będzie równy pierwszemu z tych parametrów. Taki mechanizm przekazywania wartości NaN pozwala na zidentyfikowanie funkcji, w której wystąpił błąd.

Uwaga

Jeżeli wynik jest równy NaN, nie jest wysyłany sygnał wyjściowy ok.

W zamieszczonej poniżej tabeli pokazano czy wysyłany jest sygnał wyjściowy, jeżeli jeden lub obydwa parametry są równe NaN.

Tabela E-1. Przepływ sygnału przy operacjach zmiennoprzecinkowych

Działanie	Wartość parametru 1	Wartość parametru 2	Wartość	Przepływ sygnału
Wszystkie	Poprawna liczba	Poprawna liczba	Wartość nieskończenie duża dodatnia lub ujemna	Nie
Wszystkie działania za wyjątkiem dzielenia	Nieskończoność	Poprawna liczba	Nieskończoność	Tak
Wszystkie	Poprawna liczba	Nieskończoność	Nieskończoność	Tak
Dzielenie bez reszty	Nieskończoność	Poprawna liczba	Nieskończoność	Nie
Wszystkie	Poprawna liczba	Poprawna liczba	NaN	Nie

A

ACOS, 6-10
 ADD, 6-2
 ADD_IOM, 2-24
 ADD_SIO, 2-24
 Alarmy, 3-2
 AND, 8-3
 ANY_FLT, 2-25
 APL_FLT, 2-24
 APL_FLT, 2-24
 Arcus cosinus Funkcja, 6-10
 Arcus sinus Funkcja, 6-10
 Arcus tangens Funkcja, 6-10
 ARRAY_MOVE, 10-2
 ASIN, 6-10
 ATAN, 6-10
 Awaria komunikacji podczas przesyłania programu do sterownika, 3-15

B

BAD_PWD, 2-25
 BAD_RAM, 2-24
 BCD-4, 2-22, 11-2
 BCLR, 8-14
 BIT, 2-22
 BITSEQ, 9-11
 wymagana pamięć, 9-11
 BLKCLR, 9-7
 BLKMOV, 9-5
 Blokada podglądu, 2-38
 Blokowanie/odblokowywanie podprogramów, 2-38
 Błąd w oprogramowaniu wyspecjalizowanego modułu dodatkowego, 3-10
 Błąd w programie sterującym, 3-11
 Błędna konfiguracja systemu., 3-9
 Błędna suma kontrolna programu, 3-10
 Błędne hasło dostępu do sterownika, 3-12
 Błędy diagnostyczne, 3-4
 addition of I/O module, 3-17
 application fault, 3-11
 constant sweep time exceeded, 3-11
 loss of I/O module, 3-16
 loss of, or missing, option module, 3-8
 low battery signal, 3-10
 reset of, addition of, or extra, option module, 3-8
 Błędy informacyjne, 3-4
 no user program present, 3-12
 password access failure, 3-12
 Błędy krytyczne, 3-4
 communications failure during store, 3-15
 corrupted user program on power-up, 3-12
 option module software failure, 3-10

PLC CPU system software failure, 3-13
 program block checksum failure, 3-10
 system configuration mismatch, 3-9
 Błędy operacyjne, 3-2
 Błędy wewnętrzne, 3-2
 Błędy, 3-2
 addition of I/O module, 3-17
 application fault, 3-11
 błędy działania, 3-2
 communications failure during store, 3-15
 constant sweep time exceeded, 3-11
 corrupted user program on power-up, 3-12
 dodatkowe skutki wystąpienia błędów działania, 3-5
 działania podejmowane po wystąpieniu błędu, 3-8
 grupy błędów działania sterownika, B-4
 grupy błędów działania układów wejść/wyjść, B-10
 interpretowanie błędu, B-1
 klasy błędów, 3-2
 kody błędów, B-5
 loss of I/O module, 3-16
 loss of, or missing, option module, 3-8
 low battery signal, 3-10
 no user program present, 3-12
 opis tabeli błędów działania sterownika, 3-7
 opis tabeli błędów działania układów wejść/wyjść, 3-16
 option module software failure, 3-10
 password access failure, 3-12
 PLC CPU system software failure, 3-13
 program block checksum failure, 3-10
 reakcja systemu na wystąpienie błędów, 3-3
 reset of, addition of, or extra, option module, 3-8
 system configuration mismatch, 3-9
 tabela błędów działania sterownika, 3-3, 3-5
 tabela błędów działania układów wejść/wyjść, 3-3, 3-5
 uszkodzenia wewnętrzne, 3-2
 wagi błędów działania układów wejść/wyjść, B-11
 wagi błędów, 3-4
 wagi błędów, B-5
 wyświetlanie dodatkowych informacji o błędzie, 3-6
 zewnętrzne uszkodzenia systemów wejść/wyjść, 3-2
 zmiennie, 3-4
 Błędy, interpretowanie, B-1
 BPOS, 8-16
 Brak programu sterującego, 3-12
 BSET, 8-14
 BTST, 8-12
 BYTE, 2-22

C

Call Funkcja, 12-2
 CALL, 12-2

- CFG_MM, 2-24
- COMMENT, 12-29
- COMMREQ, 9-14

- COS, 6-10
- Cosinus Funkcja, 6-10
- CTRL Klawisz, D-1
- Cykl pracy jednostki centralnej, 2-2
- Cykl pracy o stałym czasie trwania, 2-13, 2-35
- Cykl pracy sterownika, 2-2
 - cykl pracy o stałym czasie trwania, 2-13
 - czasy obsługi modułów przez jednostki centralnej serii 35x i 36x, 2-5, 2-6
 - inicjalizacja, 2-7
 - komunikacja jednostki centralnej z modułem DSM, 2-12
 - komunikacja sterownika z modułem programowalnego procesora (PCM), 2-12
 - komunikacja z programatorem, 2-9
 - obliczanie czasu trwania cyklu pracy sterownika, 2-7
 - obliczanie sumy kontrolnej programu sterującego, 2-8
 - odczyt wejść, 2-7
 - składniki czasu trwania cyklu pracy sterownika, 2-4
 - standardowy cykl pracy sterownika, 2-2
 - tryb STOP, 2-13
 - ustawianie wyjść, 2-8
 - warianty cykli pracy sterownika, 2-13
 - wykonywanie części logicznej programu sterującego, 2-8
 - wykonywanie programu sterującego, 2-8
- Cykliczne przemieszczanie w prawo lub w lewo bitu o wartości 1 Funkcja, 9-11
- Czasy obsługi modułów przez jednostki centralnej serii 35x i 36x, 2-5, 2-6
- Czasy wykonywania elementów logicznych, A-1
 - modele standardowe, A-2
 - modele zaawansowane, A-6
- SER, A-10
- Czasy wykonywania instrukcji, A-1
 - modele standardowe, A-2
 - modele zaawansowane, A-6
- SER, A-10

- D**
- Dane Globalne Ethernet, 2-43
- Dane Globalne Genius, 2-43
- Dane globalne, 2-43
- DEG, 6-14
- Diagnostyka wejść i wyjść, dane diagnostyczne, 2-42
- DINT, 2-22, 11-5
- DIV, 6-2
- DNCTR, 5-12
- Do I/O Funkcja, 12-3
 - rozszerzona funkcja DO I/O dla jednostek centralnych 331 i późniejszych, 12-7
- Dodatkowe skutki wystąpienia błędów działania, 3-5
- Dodawanie Funkcja, 6-2
- DOIO, 12-3
 - rozszerzona funkcja DO I/O dla jednostek centralnych 331 i późniejszych, 12-7
- Dołączony niezadeklarowany moduł wejść/wyjść, 3-17
- Domyślne wartości wyjść dla modułów wyjść serii 30, 2-42
- Działanie systemu, 2-1
 - cykl pracy sterownika, 2-2
 - mechanizmy zabezpieczające, 2-37
 - moduły wejść/wyjść sterowników serii 90-20, 2-39
 - moduły wejść/wyjść sterowników serii 90-30, 2-39
 - organizacja programu, typy danych i typy zmiennych, 2-17
 - rozruch i wyłączanie sterownika, 2-30
 - zegary i przełączniki czasowe, 2-34
- Dzielenie Funkcja, 6-2

- E**
- EDITLOCK, 2-38
- END, 12-21
- ENDMCR, 12-25
- EQ, 7-1
- Etykiety, 12-28
- EXP, 6-12
- EXPT, 6-12

- F**
- Format BCD
 - format zapisu czasu, 12-20
- formaty danych, 2-42
- Funkcje do przesyłania danych, 9-1
 - BITSEQ, 9-11
 - BLKCLR, 9-7
 - BLKMOV, 9-5
 - COMMREQ, 9-14
 - MOVE, 9-2
 - SHFR, 9-8
- Funkcje konwersji, 11-1
 - BCD-4, 11-2
 - DINT, 11-5
 - INT, 11-3
 - REAL, 11-7
 - TRUN, 11-11
 - WORD, 11-9
- Funkcje specjalne sterownika:
 - zmiana/ odczyt liczby słów do obliczania sumy kontrolnej, 12-42
- Funkcje sterujące, 12-1
 - CALL, 12-2
 - COMMENT, 12-29
 - DOIO, 12-3

rozszerzona funkcja DO I/O dla jednostek centralnych 331 i późniejszych, 12-7

END, 12-21

ENDMCR, 12-25

JUMP, 12-26

LABEL, 12-28

MCR, 12-22

PID, 12-71

SER, 12-8

SVCREQ, 12-30

Funkcje SVCREQ

kontrola występowania wymuszonej zmiany wartości zmiennych wejściowych i wyjściowych, 12-60

lista, 12-30

odczyt czasu trwania cyklu (#9), 12-49

odczyt czasu trwania ostatniej przerwy w zasilaniu sterownika, 12-63

odczyt nazwy folderu (#10), 12-50

odczyt nazwy sterownika (#11), 12-51

odczyt ostatnio zarejestrowanego w komunikatu o błędzie działania sterownika lub układów wejść/wyjść, 12-55

odczyt statusu trybu pracy sterownika (#12), 12-52

odczyt sumy kontrolnej programu sterującego i konfiguracji., 12-61

odczyt wartości z programatora (#2), 12-36

odczyt wskazań zegara odmierzającego czas pracy sterownika, 12-59

odczyt/ zmiana wskazań zegara czasu rzeczywistego, 12-44

pominięcie obsługi wejść i wyjść w następnym cyklu, 12-64

porównanie rzeczywistej konfiguracji modułów wejść/wyjść sterownika ze zdefiniowaną, 12-62

szybki dostęp do statusu komunikacji przez szynę komunikacyjną, 12-65

wyłączanie sterownika, 12-53

wymazanie komunikatów z tabeli błędów działania sterownika i układów wejść/wyjść, 12-54

zerowanie zegara wyłączającego (#8), 12-48

zmiana trybu komunikacji systemowe (#4), 12-40

zmiana trybu komunikacji z programatorem (#3), 12-38

zmiana/ odczyt czasu trwania cyklu pracy sterownika o stałej długości (#1), 12-33

G

GE, 7-1

Grupy błędów, B-4, B-10

GT, 7-1

H

Hasła, 2-37

HRD_CPU, 2-24

HRD_FLT, 2-25

HRD_SIO, 2-24

I

Inicjalizacja, 2-7

Instrukcje programu sterującego

konwersji, 11-1

matematyczne, 6-1

nazwy mnemoniczne instrukcji, C-1

operacje bitowe, 8-1

operacje tablicowe, 10-1

przesyłanie danych, 9-1

relacje matematyczne, 7-1

sterujące, 12-1

styki, przekaźniki i szyny, 4-1

INT, 2-22, 11-3

IO_FLT, 2-25

IO_PRES, 2-25

J

Jednostki centralnej serii 35x i 36x:

przełącznik kluczowy, 2-15

JUMP, 12-26

K

Kasowanie tabeli błędów, 12-54

Kategorie błędów, 3-16

Kody błędów, B-5

Komentarz Funkcja, 12-29

Komunikacja

komunikacja z programatorem, 2-9

Komunikacja modułu programowalnego procesora (PCM) ze sterownikiem, 2-12

Komunikacja siecią Ethernet, 2-43

Komunikacja sterownika z modułem DSM, 2-12

Komunikacja z programatorem, 2-9

Komunikacja ze sterownikiem, 2-12

Konfiguracja, 2-44

Kontrola występowania wymuszeń zmiany wartości zmiennych wejściowych i wyjściowych, 12-60

Konwersja na BCD-4 Funkcja, 11-2

Konwersja na DINT Funkcja, 11-5

Konwersja na INT Funkcja, 11-3

Konwersja na Real Funkcja, 11-7

Konwersja na Word Funkcja, 11-9

Konwersja wartości kąta Funkcja, 6-14

L

LABEL, 12-28

LE, 7-1

Liczby całkowite podwójnej precyzji ze znakiem (32 bitowe), 2-22

Liczby całkowite ze znakiem, 2-22
Liczby zmiennoprzecinkowe, E-1
 błędy w operacjach i liczbach zmiennoprzecinkowych, E-6
 wartości liczb zmiennoprzecinkowych, E-4
 wewnętrzny format reprezentacji liczb zmiennoprzecinkowych, E-3
 wprowadzanie i wyświetlanie liczb zmiennoprzecinkowych, E-5
Licznik dodający, 5-11
Licznik odejmujący, 5-12
Liczniki
 blok danych, 5-1
 DNCTR, 5-12
 UPCTR, 5-11
LN, 6-12
LOG, 6-12
Logarytm dziesiętny Funkcja, 6-12
Logarytm naturalny Funkcja, 6-12
Logarytmiczne Funkcje, 6-12
 logarytm naturalny, 6-12
 logarytm o podstawie 10, 6-12
Logiczne AND Funkcja, 8-3
Logiczne NOT Funkcja, 8-7
Logiczne OR Funkcja, 8-3
Logiczne XOR Funkcja, 8-5
Lokalne zmienne dyskretne, 2-20
LOS_IOM, 2-24
LOS_SIO, 2-24
Loss of I/O module, 3-16
Loss of, or missing, option module, 3-8
Low battery signal, 3-10
LOW_BAT, 2-24
LT, 7-1

M

Matematyczne funkcje, 6-1
 ACOS, 6-10
 ADD, 6-2
 ASIN, 6-10
 ATAN, 6-10
 COS, 6-10
 DEG, 6-14
 DIV, 6-2
 EXP, 6-12
 EXPT, 6-12
 LN, 6-12
 LOG, 6-12
 MOD, 6-6
 MUL, 6-2
 RAD, 6-14
 SIN, 6-10
 SQRT, 6-8
 SUB, 6-2
 TAN, 6-10
MCR, 12-22
Mechanizm blokowania, 2-38
 blokada edycji, 2-38
 blokada podglądu, 2-38
 stała blokada edycji i podglądu, 2-38

Mechanizmy zabezpieczające, 2-37
 blokowanie/ odblokowywanie podprogramów, 2-38
 hasła, 2-37
 poziomy dostępu, 2-37
 zmiana poziomu dostępu, 2-38
Mniejszy Funkcja, 7-1
Mniejszy lub równy Funkcja, 7-1
Mnożenie Funkcja, 6-2
MOD, 6-6
Modulo Funkcja, 6-6
Moduły wejść/ wyjść serii 90-30
 dane diagnostyczne, 2-42
 dane globalne, 2-43
 domyślne wartości wyjść dla modułów wyjść serii 30, 2-42
 formaty danych, 2-42
 zestawienie modułów wejść/wyjść serii 90-30, 2-40
Moduły wejść/wyjść dla sterowników serii 90-20, 2-39
 zestawienie modułów wejść/wyjść serii 90-20, 2-43
MOVE, 9-2
MSKCMP, 8-18
MUL, 6-2

N

Nazwy mnemoniczne, C-1
NE, 7-1
NOT, 8-7

O

Obliczanie sumy kontrolnej programu, 2-8
Obsługa błędów, 3-2
 procesor alarmów, 3-2
 wagi błędów, 3-4
Ochrona pamięci w jednostkach centralnych serii 35x i 36x, 2-15
Odczyt czasu trwania cyklu, 12-49
Odczyt identyfikatora sterownika, 12-51
Odczyt nazwy folderu, 12-50
Odczyt ostatnio zarejestrowanego komunikatu o błędzie działania, 12-55
Odczyt sumy kontrolnej programu sterującego i konfiguracji, 12-61
Odczyt trybu pracy sterownika, 12-52
Odczyt wartości z programatora , 12-36
Odczyt wejść, 2-7
Odczyt wskazań zegara czasu pracy sterownika, 12-59
Odczyt wskazań zegara czasu trwania przerwy w zasilaniu sterownika, 12-63
Odczyt wskazań zegara odmierzającego czas przerwy w zasilaniu sterownika, 2-35
Odczyt/ zmiana wskazań zegara czasu rzeczywistego, 12-44

- Odejmuwanie Funkcja, 6-2
 - OFDT, 5-8
 - ONDTR, 5-3
 - Operacje bitowe, 8-1
 - AND, 8-3
 - BCLR, 8-14
 - BPOS, 8-16
 - BSET, 8-14
 - BTST, 8-12
 - MCMP, 8-18
 - NOT, 8-7
 - OR, 8-3
 - ROL, 8-10
 - ROR, 8-10
 - SHL, 8-8
 - SHR, 8-8
 - XOR, 8-5
 - Operacje tablicowe, 10-1
 - ARRAY_MOVE, 10-2
 - poszukiwanie wartości mniejszej lub równej, 10-6
 - SRCH_GE, 10-6
 - Opis działania sterownika, cykle, 2-2
 - cykl pracy o stałym czasie trwania, 2-13, 2-35
 - czasy obsługi modułów przez jednostki centralnej serii 35x i 36x, 2-5, 2-6
 - inicjalizacja, 2-7
 - komunikacja jednostki centralnej z modułem DSM, 2-12
 - komunikacja sterownika z modułem programowalnego koprocatora (PCM), 2-12
 - komunikacja z programatorem, 2-9
 - obliczanie czasu trwania cyklu pracy sterownika, 2-7
 - obliczanie sumy kontrolnej programu sterującego, 2-8
 - odczyt wejść, 2-7
 - składniki czasu trwania cyklu pracy sterownika, 2-4
 - standardowy cykl pracy sterownika, 2-2
 - tryb STOP, 2-13
 - ustawianie wyjść, 2-8
 - warianty cykli pracy sterownika, 2-13
 - wykonywanie części logicznej programu sterującego, 2-8
 - wykonywanie programu sterującego, 2-8
 - Opis i usuwanie błędów, 3-1
 - addition of I/O module, 3-17
 - application fault, 3-11
 - błędy nie konfigurowalne, 3-8
 - communications failure during store, 3-15
 - constant sweep time exceeded, 3-11
 - corrupted user program on power-up, 3-12
 - grupy błędów działania sterownika, B-4
 - grupy błędów działania układów wejść/wyjść, B-10
 - interpretowanie błędu, B-1
 - kategoria błędu, 3-16
 - loss of I/O module, 3-16
 - loss of, or missing, option module, 3-8
 - low battery signal, 3-10
 - no user program present, 3-12
 - opis tabeli błędów działania wejść/wyjść, 3-16
 - opis tabeli błędów sterownika, 3-7
 - opisy błędów, 3-16
 - option module software failure, 3-10
 - password access failure, 3-12
 - PLC CPU system software failure, 3-13
 - program block checksum failure, 3-10
 - przetwarzanie błędów, 3-2
 - reset of, addition of, or extra, option module, 3-8
 - system configuration mismatch, 3-9
 - tabela błędów działania sterownika, 3-5
 - tabela błędów działania układów wejść/wyjść, 3-5
 - typ błędu, 3-16
 - wyświetlanie dodatkowych informacji o błędzie, 3-6
 - Opisy błędów, 3-16
 - OR, 8-3
 - Organizacja programu, typy danych/ typy zmiennych, 2-17
 - liczby zmiennoprzecinkowe, E-1
 - pamięć stanu, 2-21
 - status systemu, 2-23
 - struktura bloków funkcyjnych, 2-26
 - typy danych, 2-22
 - typy zmiennych, 2-20
 - wymuszanie wartości zmiennych, 2-21
 - OV_SWP, 2-24
 - Oznaczenia skrótowe instrukcji, C-1
- ## P
- Pamięć stanu, 2-21
 - Parametry bloku funkcyjnego, 2-28
 - PB_SUM, 2-24
 - PID, 12-71
 - Pierwiastek kwadratowy Funkcja, 6-8
 - Pionowe szyny, 4-7
 - Podprogramy
 - wywoływanie podprogramów w języku C, 2-19
 - wywoływanie, 2-19
 - Podprogramy wykonywane okresowo, 2-19
 - Podprogramy, blokowanie/ odblokowywanie, 2-38
 - Podręczniki
 - dla modułów wejść/wyjść, 2-40
 - Pominięcie obsługi wejść/wyjść w następnym cyklu, 12-64
 - Porównanie rzeczywistej konfiguracji modułów wejść/wyjść sterownika ze zdefiniowaną, 12-62
 - Porównanie z maskowaniem Funkcja, 8-18
 - POSIX Format
 - format zapisu czasu przez blok funkcyjny SER, 12-20
 - Potęga liczby e Funkcja, 6-12
 - Potęga liczby X Funkcja, 6-12
 - Poziomy dostęp, 2-37

- zmiana, 2-38
 - Procesor alarmów, 3-2
 - Program block checksum failure, 3-10
 - Programowanie, instrukcje
 - konwersji, 11-1
 - matematyczne, 6-1
 - nazwy mnemoniczne, C-1
 - operacje bitowe, 8-1
 - operacje tablicowe, 10-1
 - przesyłanie danych, 9-1
 - relacje matematyczne, 7-1
 - sterujące, 12-1
 - styki, przekaźniki i szyny, 4-1
 - Przełącznik czasowy bez pamięci z zanegowanym wejściem, 5-8
 - Przełącznik czasowy z pamięcią, 5-3, 5-5
 - Przełącznik kontynuacji, 4-8
 - Przełącznik o stykach zamkniętych, z pamięcią, 4-4
 - Przełącznik o stykach zamkniętych, 4-4
 - Przełącznik RESET z pamięcią, 4-6
 - Przełącznik RESET, 4-5
 - Przełącznik SET z pamięcią, 4-6
 - Przełącznik SET, 4-5
 - Przełącznik uaktywniany zboczem narastającym sygnału, 4-4
 - Przełącznik uaktywniany zboczem opadającym sygnału, 4-5
 - Przełącznik z pamięcią, 4-4
 - Przełączniki czasowe, 2-34
 - blok danych, 5-1
 - generator sygnału prostokątnego, 2-36
 - OFDT, 5-8
 - ONDTR, 5-3
 - TMR, 5-5
 - zegar czasu trwania przerwy w zasilaniu sterownika, 2-35
 - zegar kontrolujący czas trwania cyklu pracy sterownika w trybie o stałym czasie trwania, 2-35
 - zegar wyłączający, 2-35
 - Przełączniki, 4-2, 4-3
 - przełącznik kontynuacji, 4-8
 - przełącznik o stykach zamkniętych, 4-4
 - przełącznik o stykach zamkniętych, z pamięcią, 4-4
 - przełącznik RESET z pamięcią, 4-6
 - przełącznik RESET, 4-5
 - przełącznik SET z pamięcią, 4-6
 - przełącznik SET, 4-5
 - przełącznik uaktywniany zboczem narastającym sygnału, 4-4
 - przełącznik uaktywniany zboczem opadającym sygnału, 4-5
 - przełącznik z pamięcią, 4-4
 - Przekroczony czas trwania cyklu sterownika, 3-11
 - Przełącznik kluczowy w jednostkach centralnych serii 35x 36x, 2-15
 - Przemieszczanie bloku pamięci Funkcja, 9-2
 - Przeptyw sygnału, 2-29
 - Przesunięcie słowa bitowego w lewo
 - Funkcja, 8-8
 - Przesunięcie słowa bitowego w lewo w obiegu zamkniętym Funkcja, 8-10
 - Przesunięcie słowa bitowego w prawo
 - Funkcja, 8-8
 - Przesunięcie słowa bitowego w prawo w obiegu zamkniętym Funkcja, 8-10
 - Przeszukiwanie tablicy Funkcja, 10-2
 - Przybliżanie wartości liczby Funkcja, 11-11
 - Przykłady
 - SER, 12-16
- ## R
- RAD, 6-14
 - RANGE, 7-4
 - REAL
 - konwersja na REAL, 11-7
 - liczby rzeczywiste, E-1
 - liczby zmiennoprzecinkowe, E-1
 - struktura typów danych, 2-22
 - Regulator proporcjonalno- całkująco- różniczkujący (PID), 12-71
 - Rejestry, 2-20
 - wejścia analogowe, 2-20
 - wyjścia analogowe, 2-20
 - Relacje matematyczne, 7-1
 - EQ, 7-1
 - GE, 7-1
 - GT, 7-1
 - LE, 7-1
 - LT, 7-1
 - NE, 7-1
 - RANGE, 7-4
 - Reset of, addition of, or extra, option module, 3-8
 - ROL, 8-10
 - ROR, 8-10
 - Rozładowane baterie, 3-10
 - Rozruch i wyłączanie sterownika, 2-30
 - rozruch, 2-30
 - wyłączanie, 2-33
 - Rozruch sterownika, 2-30
 - Rozszerzona funkcja DO I/O dla jednostek centralnych 331 i późniejszych, 12-7
 - Równy Funkcja, 7-1
 - Różny Funkcja, 7-1
- ## S
- Sequential Event Recorder, 12-9.
 - SER Funkcja, 12-8
 - dane diagnostyczne, 2-42
 - dane globalne, 2-43
 - domyślne wartości wyjść dla modułów wyjść serii 30, 2-42
 - formaty danych, 2-42
 - moduły wejść/wyjść sterowników 90-30, 2-40

- SFT_CPU, 2-25
- SFT_FLT, 2-25
- SFT_SIO, 2-24
- SHFR Funkcja, 9-8
- SHFR, 9-8
- SHL, 8-8
- SHR, 8-8
- SIN, 6-10
- Sinus Funkcja, 6-10
- Skróty klawiaturowe, D-1
- SNPX_RD, 2-24
- SNPX_WT, 2-24
- SNPXACT, 2-24
- Sprawdzanie wartości bitu Funkcja, 8-12
- SQRT, 6-8
- SRCH_GE, 10-6
- SRCH_LE, 10-6
- Standardowy cykl pracy sterownika, 2-2
- STOR_ER, 2-25
- Struktura bloku funkcyjnego, 2-26
 - format bloków funkcyjnych, 2-26
 - format przekaźników, 2-26
 - parametry bloków funkcyjnych, 2-28
 - sygnał sterujący, 2-29
- Struktura programu
 - wywoływanie bloków, 2-19
 - wywoływanie podprogramów w języku C, 2-19
 - wywoływanie podprogramów, 2-19
- struktura wejść i wyjść sterownika serii 90-30, 2-39

- Styk kontynuacji, 4-8
- Styk otwarty, 4-3
- Styk zamknięty, 4-3
- Styki generatora sygnału prostokątnego, 2-36
- Styki, 4-1
 - styk kontynuacji, 4-8
 - styk otwarty, 4-3
 - styk zamknięty, 4-3
- Styki, przekaźniki i szyny, 4-1
 - przekaźnik kontynuacji, 4-8
 - przekaźnik o stykach zamkniętych, 4-4
 - przekaźnik o stykach zamkniętych, z pamięcią, 4-4
 - przekaźnik RESET z pamięcią, 4-6
 - przekaźnik RESET, 4-5
 - przekaźnik SET z pamięcią, 4-6
 - przekaźnik SET, 4-5
 - przekaźnik uaktywniany zboczem narastającym sygnału, 4-4
 - przekaźnik uaktywniany zboczem opadającym sygnału, 4-5
 - przekaźnik z pamięcią, 4-4
 - przekaźniki, 4-2, 4-3
 - styk kontynuacji, 4-8
 - styk otwarty, 4-3
 - styk zamknięty, 4-3
 - styki, 4-1
 - szyny poziome i pionowe, 4-7

- SUB, 6-2
- SY_FLT, 2-25
- SY_PRES, 2-25
- System configuration mismatch, 3-9
- System wejść/wyjść sterowników serii 90/30, 2-39
- Systemowe zmienne rejestrowe, 2-20
- Systemowe zmienne stanu, 2-21, 2-23
 - ADD_IOM, 2-24
 - ADD_SIO, 2-24
 - ANY_FLT, 2-25
 - APL_FLT, 2-24
 - BAD_PWD, 2-25
 - BAD_RAM, 2-24
 - CFG_MM, 2-24
 - HRD_CPU, 2-24
 - HRD_FLT, 2-25
 - HRD_SIO, 2-24
 - IO_FLT, 2-25
 - IO_PRES, 2-25
 - LOS_IOM, 2-24
 - LOS_SIO, 2-24
 - LOW_BAT, 2-24
 - OV_SWP, 2-24
 - PB_SUM, 2-24
 - SFT_CPU, 2-25
 - SFT_FLT, 2-25
 - SFT_SIO, 2-24
 - SNPX_RD, 2-24
 - SNPX_WT, 2-24
 - SNPXACT, 2-24
 - STOR_ER, 2-25
 - SY_FLT, 2-25
 - SY_PRES, 2-25
- Szukanie elementu mniejszego lub równego, 10-6
- Szukanie elementu większego lub równego Funkcja, 10-6
- Szybki dostęp do statusu komunikacji przez szynę komunikacyjną, 12-65
- Szyny poziome i pionowe, 4-7
- Szyny, 4-7

- T**
- Tabela błędów działania sterownika, 3-3, 3-15, B-1
 - data i czas wystąpienia błędu, B-7
 - długość bloku dodatkowych informacji o błędzie, B-3
 - dodatkowe informacje o błędzie, B-7
 - gniazdo, B-3
 - grupy błędów, B-4
 - interpretacja, 3-7
 - interpretowanie błędu, B-1
 - kaseta, B-3
 - kody błędów, B-5
 - wagi błędów, B-5
 - zadanie, B-3

- Tabela błędów działania układów
wejść/wyjść, 3-3, 3-5,
B-8
adres wystąpienia błędu, B-9
adres zmiennej, B-9
blok szczegółowych informacji o błędzie,
B-11
data i czas wystąpienia błędu, B-12
długość bloku dodatkowych informacji
o błędzie, B-9
gniazdo, B-10
grupy błędów, B-10
interpretacja, 3-16
interpretowanie błędu, B-1
kasety, B-10
punkt, B-10
symboliczne informacje o błędach działania,
B-11
wagi błędów a działania realizowane przez
sterownik,
B-11
wagi błędów, B-11
- TAN, 6-10
Tangens Funkcja, 6-10
TMR, 5-5
TRUN, 11-11
tryb STOP, 2-13
Tryby komunikacji, 2-14
Typ błędu, 3-16
Typy danych, 2-22
BCD-4, 2-22
BIT, 2-22
BYTE, 2-22
DINT, 2-22
INT, 2-22
REAL, 2-22
WORD, 2-22
- Typy zmiennych, 2-20
dane globalne, 2-21
lokalne zmienne dyskretne, 2-20
rejstry systemowe, 2-20
stanu systemu, 2-21, 2-23
wejścia analogowe, 2-20
wejścia dyskretne, 2-20
wewnętrzne zmienne dyskretne, 2-20
wyjścia analogowe, 2-20
wyjścia dyskretne, 2-20
zmienne dyskretne, 2-20
zmienne rejestrowe, 2-20
zmienne systemowe, 3-4
- U**
- UPCTR, 5-11
Ustawianie wartości bitu na 1 Funkcja, 8-16
Ustawianie wartości pojedynczego bitu na 0
Funkcja, 8-14
Ustawianie wyjść, 2-8
Usterki zewnętrznych modułów wejść/wyjść,
3-2
Usuwanie zakłóceń, 3-1
błędy nie konfigurowalne, 3-8
- interpretowanie błędu, B-1
opis tabeli błędów działania sterownika, 3-7
opis tabeli błędów działania układów
wejść/wyjść, 3-16
tabela błędów działania sterownika, 3-5
tabela błędów działania układów wejść/wyjść
, 3-5
wyświetlanie dodatkowych informacji
o błędzie, 3-6
- Uszkodzenie pamięci, 3-7
- W**
- Wagi błędów a działania podejmowane w
momencie ich wystąpienia, 3-8
Wagi błędów, 3-4
błędy diagnostyczne, 3-4
błędy informacyjne, 3-4
błędy krytyczne, 3-4
wagi błędów działania sterownika, B-5
wagi błędów działania układów wejść/wyjść,
B-11
- Warianty cykli pracy sterownika, 2-13
Wewnętrzne zmienne dyskretne, 2-20
Większy Funkcja, 7-1
Większy lub równy Funkcja, 7-1
WORD, 2-22, 11-9
Wykładnicze Funkcje, 6-12
potęga liczby e, 6-12
potęga X, 6-12
- Wykonywanie programu sterującego, 2-8
Wykrycie uszkodzenia programu sterującego
podczas rozruchu, 3-12
Wyłączanie sterownika, 2-33
Wymuszone zmiany wartości zmiennych,
2-21
- X**
- XOR, 8-5
- Z**
- Zagnieżdżona forma instrukcji ENDMCR,
12-25
Zagnieżdżona forma instrukcji MCR, 12-22
Zatrzymywanie sterownika, Blok funkcyjny,
12-53
Zegar kontrolujący czas trwania cyklu pracy
sterownika w trybie ze stałym czasem
cyklu, 2-35
Zegar odmierzający czas pracy sterownika,
2-34
zegar podtrzymujący aktualną datę i czas,
2-34
Zegar wyłączający, 2-35
Zegary, 2-34
zegar odmierzający czas pracy sterownika,
2-34

zegar podtrzymujący aktualną datę i czas,
2-34

Zerowanie bloku pamięci Funkcja, 9-7

Zestaw instrukcji

- konwersji Funkcje, 11-1
- matematyczne Funkcje, 6-1
- operacje bitowe, 8-1
- operacje tablicowe, 10-1
- przesyłanie danych Funkcje, 9-1
- relacje matematyczne, 7-1
- sterujące Funkcje, 12-1
- styki, przekaźniki i szyny, 4-1

Zmiana poziomu dostępu, 2-38

Zmiana trybu i czasu komunikacji z
programatorem, 12-38

Zmiana trybu i czasu trwania komunikacji
systemowej, 12-40

Zmiana/ odczyt czasu trwania cyklu pracy
sterownika w trybie o stałym czasie
trwania, 12-33

Zmiana/ odczyt liczby słów do obliczania
sumy kontrolnej, 12-42

Zmienne do obsługi błędów, 3-4

- definicje, 3-4

Zmienne dyskretne, 2-20

- dane globalne, 2-21
- lokalne zmienne dyskretne, 2-20
- stanu systemu, 2-21, 2-23
- wejścia dyskretne, 2-20
- wewnętrzne zmienne dyskretne, 2-20
- wyjścia dyskretne, 2-20
- zmienne systemowe, 3-4

Zmienne globalne, 2-21

Zmienne przypisane wejściom analogowym,
2-20

Zmienne przypisane wejściom dyskretnym,
2-20

Zmienne przypisane wyjściom analogowym,
2-20

Zmienne przypisane wyjściom dyskretnym,
2-20

Zmienne rejestrowe

- rejstry systemowe, 2-20

Zmienne systemowe, 2-21, 2-23, 3-5

Zmienne, 2-20